Deep Neural Networks Guided Ensemble Learning for Point Estimation in Finite Samples

Tianyu Zhan*
Data and Statistical Sciences, AbbVie Inc., North Chicago, IL, USA

Haoda Fu†
Department of Advanced Analytics and Data Sciences,
Eli Lilly and Company, Indianapolis, IN, USA

Jian Kang‡
Department of Biostatistics, University of Michigan, Ann Arbor, MI, USA

May 17, 2021

Abstract

As one of the most important estimators in classical statistics, the uniformly minimum variance unbiased estimator (UMVUE) has been adopted for point estimation in many statistical studies, especially for small sample problems. Moving beyond typical settings in the exponential distribution family, it is usually challenging to prove

*Tianyu Zhan is an employee of AbbVie Inc. Corresponding author email address: tianyu.zhan.stats@gmail.com.

†Haoda Fu is an employee of Eli Lilly and Company. Email address: fu_haoda@lilly.com.

‡Jian Kang is Professor in the Department of Biostatistics at the University of Michigan, Ann Arbor. Kang's research was partially supported by NIH R01 GM124061 and R01 MH105561. Email address: jiankang@umich.edu.

the existence and further construct such UMVUE in finite samples. For example in the ongoing Adaptive COVID-19 Treatment Trial (ACTT), it is hard to characterize the complete sufficient statistics of the underlying treatment effect due to pre-planned modifications to design aspects based on accumulated unblinded data. As an alternative solution, we propose a Deep Neural Networks (DNN) guided ensemble learning framework to construct an improved estimator from existing ones. We show that our estimator is consistent and asymptotically reaches the minimal variance within the class of linearly combined estimators. Simulation studies are further performed to demonstrate that our proposed estimator has considerable finite-sample efficiency gain. In the ACTT on COVID-19 as an important application, our method essentially contributes to a more ethical and efficient adaptive clinical trial with fewer patients enrolled.

Keywords: Adaptive COVID-19 Treatment Trial (ACTT); Consistent estimation; Deep learning; Efficiency

1 Introduction

Identifying the potential uniformly minimum variance unbiased estimator (UMVUE) of an unknown parameter is one of the most fundamental and important problems in statistics. It provides uniformly no larger variance than any other unbiased estimators in the parameter space considered (Lehmann and Casella, 2006). However, its existence and characterization are usually challenging to investigate when one moves beyond exponential families. For example in the ongoing Adaptive COVID-19 Treatment Trial (ACTT), adaptive clinical trials are appealing to accommodate uncertainty with limited knowledge of the treatment profiles by allowing prospectively planned modifications to design aspects based on accumulated unblinded data (Bretz et al., 2009; Chen et al., 2010, 2014; National Institutes of Health, 2020a). One is interested in an unbiased estimator of the underlying treatment effect to have an accurate assessment of the efficacy of the study drug, but traditional estimators are often biased (Bretz et al., 2009). Although several methods (Shen, 2001; Stallard et al., 2008) have been proposed to estimate the bias, its correction in adaptive design is still a less well-studied phenomenon, as acknowledged by the Food and Drug Administration [FDA; Food and Drug Administration (2019)] and the European Medicines Agency [EMA; European Medicines Agency (2007)]. Moving beyond, the next question is how to identify a more efficient unbiased estimator of the treatment effect, which further contributes to a more ethical and efficient adaptive clinical trial with fewer patients enrolled.

Since the complete sufficient statistics can be hard to characterize or do not even exist in many problems (Lehmann and Casella, 2006), we consider an alternative perspective by constructing a better estimator in finite samples from existing unbiased estimators. This is motivated by the spirit of ensemble learning to build a prediction model by combining the strengths of a collection of simpler base models (Biau, 2012; Bradic et al., 2016; Katzfuss

et al., 2016; McDermott and Wikle, 2017; Biau et al., 2019; Tian and Feng, 2021). For example, the XGBoost algorithm (Chen and Guestrin, 2016) is one of the most powerful algorithms in machine learning literature as a scalable end-to-end tree boosting system (Wang et al., 2020).

In this article, we propose a novel Deep Neural Networks (DNN) guided ensemble learning framework to provide point estimation on the parameters of interest. DNN is becoming more popular in biomedical fields in recent years due to its strong functional representation (She et al., 2014; Brahma et al., 2015; Liang et al., 2018; Lu et al., 2018; Rava and Bradic, 2020; Bai et al., 2020; Chao et al., 2020; Chen et al., 2020; Wu et al., 2020; Yuan et al., 2020; Gao and Wang, 2021). In this article, we leverage DNN to approximate the optimal weights of linearly integrating unbiased estimators to achieve minimum variance in finite sample size. We show that the bias of our estimator is asymptotically zero, and the mean squared error (MSE) converges to the optimal variance within the class of linearly combined estimators. Simulations demonstrate that our proposed estimator achieves considerable finite-sample efficiency gain, for example in the scale-uniform distribution considered in Section 5.1 where the Cramér–Rao bound is not satisfied, and in the regression model for analyzing heterogeneous data in Section 5.2. We further apply our method to the ACTT on COVID-19 to provide a more accurate estimate on the underlying treatment effect and to consistently achieve higher power of detecting a promising treatment effect than several alternatives in the context of hypothesis testing. This makes our method appealing in practice, because a more ethical trial with fewer patients enrolled can be implemented to deliver a safe and efficacious drug to patients more efficiently.

The remainder of this article is organized as follows. In Section 2, we introduce our framework of constructing an ensemble estimator with improved efficiency. Next we propose

an algorithm based on DNN to approximate the optimal weight parameters in Section 3. In Section 4, we provide the upper bounds of the bias and the MSE of our estimator. Three experiments including the ACTT on COVID-19 are conducted in Section 5 to demonstrate our superior finite sample performance. Concluding remarks are provided in Section 6.

2 An ensemble estimator

Our parameter of interest is θ under an open and bounded $\Theta \subseteq \mathbb{R}$. For illustration, θ is considered as a scalar quantity, but our proposed method can be readily applied to a vector as considered in the regression problem at Section 5.2. Let $\boldsymbol{x} = (x_1, \cdots, x_n)$ be independent and identically distributed (i.i.d.) random variables given on the probability space $(\Omega_x, \mathcal{A}_x, P_x)$, where Ω_x is a compact set in $\mathbb{R}$ and $P_x = p(x; \theta, \boldsymbol{\omega})$ is the probability function. The nuisance parameters $\boldsymbol{\omega}$ is of $d-1$ dimension with an open and bounded support $\boldsymbol{\Omega} \subseteq \mathbb{R}^{d-1}$ and d is an integer larger than 1.

An estimator $T(\boldsymbol{x})$ of θ is unbiased if

$$E\left[T(\boldsymbol{x})\right] = \theta, \tag{1}$$

for all $\theta \in \Theta$ (Lehmann and Casella, 2006). Without being further specified, the expectation $E(\cdot)$ is with respect to P_x. If there exists such an unbiased estimator $T(\boldsymbol{x})$ satisfying (1), then the estimand θ is U-estimable. An unbiased estimator is the uniformly minimum variance unbiased estimator (UMVUE) if it has no larger variance than any other unbiased estimators of θ for all $\theta \in \Theta$ (Lehmann and Casella, 2006). Despite attractive features of UMVUE, the existence and characterization are usually challenging to investigate when one moves beyond exponential families.

In this article, we propose an alternative approach to construct an improved unbiased estimator with smaller variance by ensembling two existing ones via Deep Neural Networks (DNN). Let $T_1(\boldsymbol{x})$ and $T_2(\boldsymbol{x})$ be two unbiased estimators of θ. We construct $U(\boldsymbol{x})$ by a linear combination of them,

$$U(\boldsymbol{x}; w) = w \times T_1(\boldsymbol{x}) + (1 - w) \times T_2(\boldsymbol{x}), \tag{2}$$

where $w \in \mathbb{R}$. The optimal weight $w^{\{opt\}}$ is the one that minimizes the variance of $U(\boldsymbol{x}; w)$ for $w \in \mathbb{R}$,

$$w^{\{opt\}} = \text{argmin}_{w \in \mathbb{R}} var\big[U(\boldsymbol{x}; w)\big] = \frac{E\big[\,\{T_2(\boldsymbol{x}) - T_1(\boldsymbol{x})\}\, T_2(\boldsymbol{x})\big]}{E\big[\,\{T_1(\boldsymbol{x}) - T_2(\boldsymbol{x})\}^2\,\big]}. \tag{3}$$

Since both $T_1(\boldsymbol{x})$ and $T_2(\boldsymbol{x})$ are unbiased for θ and $w^{\{opt\}}$ is a constant with respect to observed data $\boldsymbol{x}$, then $U\left(\boldsymbol{x}; w^{\{opt\}}\right)$ is also unbiased. The construction in (3) ensures that $U\left(\boldsymbol{x}; w^{\{opt\}}\right)$ has the smallest variance among all $U(\boldsymbol{x}; w)$ in (2) for $w \in \mathbb{R}$. We provide the variance reduction in the following Proposition 1 with proof in the Supplemental Materials Section 1. For simplicity, "$(\boldsymbol{x})$" is removed from the notations of $T_1(\boldsymbol{x})$ and $T_2(\boldsymbol{x})$.

Proposition 1 *The variance reduction $\Lambda(w)$ of estimating θ by $U\left(\boldsymbol{x}; w^{\{opt\}}\right)$ with $w^{\{opt\}}$ in (3) as compared with $U(\boldsymbol{x}; w)$ in (2) is*

$$\begin{aligned}\Lambda(w) &= var\left\{U(\boldsymbol{x}; w)\right\} - var\left\{U\left(\boldsymbol{x}; w^{\{opt\}}\right)\right\} \\ &= \frac{\left[E\left\{(T_2 - T_1)\, T_2\right\} - E\left\{(T_1 - T_2)^2\right\} w\right]^2}{E\left\{(T_1 - T_2)^2\right\}}.\end{aligned} \tag{4}$$

Note that this variance improvement $\Lambda(w)$ is non-negative with $\Lambda\left(w^{\{opt\}}\right) = 0$. In some

problems where (3) is free from θ and $\boldsymbol{\omega}$ or can be evaluated in a closed form, the solution of $w^{\{opt\}}$ is straightforward – for example, on estimating the mean of a normal distribution with known coefficient of variation based on two unbiased estimators from the sample mean and the sample variance (Khan, 2015). In general, $w^{\{opt\}}$ in (3) is a function of θ, $\boldsymbol{\omega}$ and sample size n, but does not necessarily have an analytic solution. For many problems, we do not have closed forms of the distributions of T_1 and T_2, and thus the direct computation is not feasible. For some other problems, T_1 and T_2 themselves do not have closed forms, and making the computation even harder.

While it is usually feasible to empirically estimate $w^{\{opt\}}$ given underlying θ and $\boldsymbol{\omega}$, our goal is to estimate $w^{\{opt\}}$ and further construct improved statistics based on observed data $\boldsymbol{x}$ with given sample size n. We further denote $\boldsymbol{\phi} = (\theta, \boldsymbol{\omega})$. In the next Section 3, we introduce our proposed algorithm for approximating $w^{\{opt\}}(\boldsymbol{\phi})$ by DNN from $\boldsymbol{x}$.

3 A DNN-based algorithm to approximate $w^{\{opt\}}$

We first provide a short review on Deep Neural Networks (DNN) in Section 3.1, and then demonstrate in Section 3.2 that there exists a DNN structure which can well approximate the underlying $w^{\{opt\}}(\boldsymbol{\phi})$ to a desired level of accuracy. In the next section 3.3, we illustrate our DNN-based algorithm to estimate $w^{\{opt\}}(\boldsymbol{\phi})$.

3.1 Review on Deep Neural Networks (DNN)

Deep learning is a specific subfield of machine learning with a major application to approximate a function $y = w(\boldsymbol{\phi})$ (Goodfellow et al., 2016). We restrict our attention to the so-called deep feedforward networks or feedforward neural networks, which define a

mapping function $y = \widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta})$ and learn the value of parameters $\boldsymbol{\eta}$ that result in the best function approximation, where $\boldsymbol{\eta}$ denotes a stack of the weights and bias parameters in the neural networks with dimension d_η.

In earlier years, it has been shown that a shallow neural network with sigmoid activation function can approximate any continuous function to a desired accuracy with sufficiently large number of nodes (Cybenko, 1989), and then interest shifted towards deeper networks with a better generalizability (Liang et al., 2018; Lu et al., 2018; Chen et al., 2020; Rava and Bradic, 2020; Wu et al., 2020). The upper bounds were also investigated on the approximation error of Lipschitz-continuous functions (Bach, 2017; Xu and Wang, 2018; Chen et al., 2019), and functions in Sobolev spaces (Yarotsky, 2017).

Consider a motivating example of a DNN with two hidden layers in Figure 1. The input parameter $\boldsymbol{\phi}$ has a dimension $d = 2$ on the left, with a scaler output y on the right. We follow the notations in Anthony and Bartlett (2009) to characterize the complexity of a DNN structure. In this simple architecture, there are 6 computation units from the two hidden layers, a total of 18 weights parameters, and 7 bias parameters. Therefore, the dimension of $\boldsymbol{\eta}$ is $d_\eta = 25$. We further define $n^{(l)}$ as the depth of DNN and $n^{(w)}$ as the total number of computation unites, weights and bias parameters. In Figure 1 we have $n^{(l)} = 4$ and $n^{(w)} = 31$.

3.2 Approximation error bound of DNN

We utilize DNN to construct a mapping function $\widetilde{w} : \boldsymbol{\Phi} \to \mathbb{R}$ to approximate $w^{\{opt\}}$, where $\boldsymbol{\phi} \in \boldsymbol{\Phi} \subseteq \mathbb{R}^d$. Before studying the approximation error, we first list the following regularity conditions,

A.1 Let $\boldsymbol{\Phi}$ of dimension d be open and bounded, with $\partial\boldsymbol{\Phi}$ of class C^1.

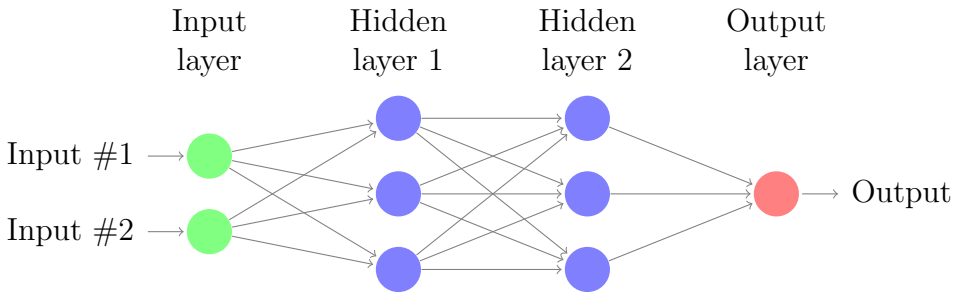

Figure 1: An illustrative Deep Neural Networks with two hidden layers.

A.2 $E\left\{(T_1)^2;\boldsymbol{\phi}\right\}$, $E\left\{(T_2)^2;\boldsymbol{\phi}\right\}$ and $E\left(T_1T_2;\boldsymbol{\phi}\right)$ are Lipschitz continuous on $\boldsymbol{\Phi}$ for some constants c_1, c_2 and c_{12}, respectively.

A.3 T_1 and T_2 have finite second moments bounded by b_1 and b_2, respectively, for $\boldsymbol{\phi} \in \boldsymbol{\Phi}$.

A.4 $\inf_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} E\left\{(T_1 - T_2)^2;\boldsymbol{\phi}\right\} \geq c_L$, for a positive constant c_L.

Remarks: Condition A.1 specifies that the parameter space $\boldsymbol{\Phi}$ is open and bounded with a continuously differentiable boundary (Evans, 2010). Condition A.2 requires that the second moments of T_1, T_2 cannot be too steep. A function $u : U \to \mathbb{R}$ is Lipschitz continuous on U if

$$|u(x) - u(y)| \leq C\,|x - y|\,,$$

for some constant C and every $x, y \in U$. This condition is weaker than differentiation but stronger than continuity. Consider an example where $\boldsymbol{x}$ of size n follow a normal distribution with mean zero and variance σ^2, and T_1 is the sample mean with $E\left\{(T_1)^2\right\} = \sigma^2/n$. It can be shown that $C = 1/n$ satisfies the above definition for every $\sigma^2 \in U \subseteq \mathbb{R}^+$. This condition is usually satisfied by T_1 and T_2 in common statistical models. These two base statistics are required to have finite second moments in Condition A.3. The fourth condition A.4 requires that the variance of $T_1 - T_2$ is lower bounded by a positive constant. A trivial

counterexample is that the variance of $T_1 - T_2$ becomes zero when $T_1 = T_2$. We provide more discussion on how to choose T_1 and T_2 in practice in Section 4.2.

In the following Proposition 2, we show that under those four regularity conditions, there exists a DNN $\widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta}_0)$ with finite $n^{(l)}$ and $n^{(w)}$ that can well approximate $w^{\{opt\}}$ with the uniform maximum error defined by,

$$\left\| w^{\{opt\}} - \widetilde{w} \right\|_\infty = \max_{\boldsymbol{\phi} \in \boldsymbol{\Phi}} \left| w^{\{opt\}}(\boldsymbol{\phi}) - \widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta}_0) \right|. \tag{5}$$

Proposition 2 *Under regularity conditions A.1 - A.4, for a given dimension d and an error tolerance $\epsilon_d \in (0, 1)$, there exists a DNN $\widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta}_0)$ with underling $\boldsymbol{\eta}_0$ and ReLU activation function that is capable of expressing $w^{\{opt\}}$ with the uniform maximum error*

$$\left\| w^{\{opt\}} - \widetilde{w} \right\|_\infty \leq \epsilon_d.$$

The DNN has a finite number of layers $n^{(l)}$, finite total number of computation unites, weight and bias parameters $n^{(w)}$, which satisfy

$$n^{(l)} < c(d) \left\{ \ln(1/\epsilon_d) + 1 \right\},$$
$$n^{(w)} < c(d) \epsilon_d^{-d} \left\{ \ln(1/\epsilon_d) + 1 \right\},$$

for some constant $c(d)$ depending on d.

The proof is provided in the Supplemental Materials Section 2. Our contribution is to show that the objective function $w^{\{opt\}}$ in (3) is Lipschitz continuous for $\boldsymbol{\phi} \in \boldsymbol{\Phi}$ under those four regularity conditions. Therefore, it belongs to a Sobolev space $W^{1,\infty}(\boldsymbol{\Phi})$ with

the norm

$$\|w\|_{W^{1,\infty}(\mathbf{\Phi})} = \max_{\boldsymbol{m}:|\boldsymbol{m}|\leq 1} \operatorname*{ess\,sup}_{\boldsymbol{\phi}\in\mathbf{\Phi}} |D^{\boldsymbol{m}} w(\boldsymbol{\phi})|\,, \tag{6}$$

where $\boldsymbol{m} = (m_1, ..., m_d) \in \{0,1\}^d$, $|\boldsymbol{m}| = \sum_{i=1}^d m_i$, $D^{\boldsymbol{m}}$ is the respective weak derivative, and "ess sup" is the essential supremum (Evans, 2010). The norm $\|w\|_{W^{1,\infty}(\mathbf{\Phi})}$ in (6) is denoted as c_d. Furthermore, the upper bounds on $n^{(l)}$ and $n^{(w)}$ of approximating functions in Sobolev spaces are obtained from Theorem 1 in Yarotsky (2017).

3.3 A DNN-based algorithm

In the previous section, we have shown that there exists a DNN $\widetilde{w}(\boldsymbol{\phi};\boldsymbol{\eta}_0)$ that can well approximate $w^{\{opt\}}(\boldsymbol{\phi})$ with a controlled uniform maximum error in (5) in Proposition 2. The next question is how to estimate $\boldsymbol{\eta}_0$ by $\widehat{\boldsymbol{\eta}}$ to construct a learnable DNN $\widetilde{w}(\boldsymbol{\phi};\widehat{\boldsymbol{\eta}})$ using data.

At Step 1 of Algorithm 1, we construct input data of DNN as $\{\boldsymbol{\phi}_m\}_{m=1}^M$, and the output label as $\{\widehat{w}(\boldsymbol{\phi}_m)\}_{m=1}^M$ of size M. The input $\{\boldsymbol{\phi}_m\}_{m=1}^M$ are i.i.d. random variables defined on a working probability space $(\mathbf{\Phi}, \boldsymbol{\mathcal{A}}_\phi, \boldsymbol{P}_\phi)$, where $\mathbf{\Phi}$ is a compact set in $\mathbb{R}^d$. The working multivariate probability function $\boldsymbol{P}_\phi$ is usually set as some flat distributions to let simulated $\{\boldsymbol{\phi}_m\}_{m=1}^M$ spread within the support $\mathbf{\Phi}$. In the remainder of this article, we draw each of the d elements in $\boldsymbol{\phi}_m$ for $m = 1, \cdots, M$ from d separate uniform distributions under its corresponding support in $\mathbf{\Phi}$. The output label is $\widehat{w}(\boldsymbol{\phi}_m)$ as an estimate of the underlying $w^{\{opt\}}(\boldsymbol{\phi}_m)$, whose functional form is usually unknown. It can be obtained from the numerical integration method if the joint distribution of T_1 and T_2 is known, or it can be estimated by the sparse grid method in a high-dimensional setting (Shen and Yu, 2010; Zhang et al., 2015), or by Monte Carlo samples. For a general demonstration, we obtain

$\widehat{w}(\boldsymbol{\phi}_m)$ with

$$\widehat{w}(\boldsymbol{\phi}_m) = \frac{\sum_{i=1}^{N} \left[\{T_2(\boldsymbol{x}_i) - T_1(\boldsymbol{x}_i)\} T_2(\boldsymbol{x}_i) \right]}{\sum_{i=1}^{N} \left[\{T_1(\boldsymbol{x}_i) - T_2(\boldsymbol{x}_i)\}^2 \right]}, \tag{7}$$

where $\boldsymbol{x}_i$ of size n are drawn from the distribution function $p(x; \boldsymbol{\phi}_m)$, for $i = 1, \cdots, N$.

Given the underlying parameters $\boldsymbol{\phi}_m$, it is usually feasible to compute $\widehat{w}(\boldsymbol{\phi}_m)$ in (7) as a consistent estimator of $w^{\{opt\}}(\boldsymbol{\phi}_m)$ in (3). However, one is more interested in estimating $w^{\{opt\}}(\boldsymbol{\phi}_m)$ based on observed data $\boldsymbol{x}$. The true model is

$$\widehat{w}(\boldsymbol{\phi}_m) = w^{(opt)}(\boldsymbol{\phi}_m) + e_m, \tag{8}$$

and the working model is

$$\widehat{w}(\boldsymbol{\phi}_m) = \widetilde{w}(\boldsymbol{\phi}_m; \boldsymbol{\eta}_0) + \widetilde{e}_m, \tag{9}$$

where e_m converges in probability to zero as M goes to infinity, and $\widetilde{e}_m$, for $m = 1, \cdots, M$, are assumed to be i.i.d. random errors with zero mean and finite variance.

In statistics, fitting a neural network can be viewed as a nonlinear regression problem to find the least squared estimator $\widehat{\boldsymbol{\eta}}$ of $\boldsymbol{\eta}_0$ (White, 1990; Shen et al., 2019), where $\widehat{\boldsymbol{\eta}}$ is given by

$$\widehat{\boldsymbol{\eta}} = \underset{\boldsymbol{\eta} \in \boldsymbol{H}}{\arg\min} \frac{1}{M} \sum_{m=1}^{M} \left\{ \widehat{w}(\boldsymbol{\phi}_m) - \widetilde{w}(\boldsymbol{\phi}_m; \boldsymbol{\eta}) \right\}^2, \tag{10}$$

and $\boldsymbol{H}$ is a compact subset of $\mathbb{R}^{d_\eta}$; and recall that d_η is the dimension of $\boldsymbol{\eta}$. There are many challenges in obtaining $\widehat{\boldsymbol{\eta}}$ and further studying its properties. The structure of DNN $\widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta}_0)$ in (9) which satisfies the approximation error bound in Proposition 2 is usually unknown. The identifiability of $\boldsymbol{\eta}_0$ is questionable if the structure of $\widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta})$ in (9) is arbitrarily complex (White, 1990). Shen et al. (2019) considered a sieve as a sequence of function classes indexed by size M, and further established the consistency and asymptotic

properties of a least squared estimator under sub-Gaussian errors $\widetilde{e}_m$ in (9) with one hidden layer and sigmoid activation function. Furthermore, since the loss function in (10) is usually non-convex, identifying $\widehat{\boldsymbol{\eta}}$ is a challenging and active field in machine learning (Goodfellow et al., 2016). We utilize the RMSProp for the DNN fitting at Step 2 and 3, as it has been shown to be an effective and practical optimization algorithm (Hinton et al., 2012; Goodfellow et al., 2016).

It is important to select a proper DNN structure by cross-validation at Step 2 (Goodfellow et al., 2016). By increasing the number of layers and number of nodes in DNN, the empirical MSE from the training dataset usually decreases by containing more complex structures. However, the MSE in the validation dataset or the MSE from the Jackknife method is subject to increasing with poor performance at generalization tasks. Then one can further implement certain regulation approaches, for example dropout techniques or L_1 and L_2 regularization methods, on the over-saturated DNN structure to decrease validation MSE while keeping the training MSE below a certain tolerance, for example 10^{-5}. Several candidates around this sub-optimal structure can be proposed, and the final structure at Step 3 is selected by cross-validation with the smallest validation MSE from this candidate pool to obtain the fitted DNN $\widetilde{w}(\boldsymbol{\phi}; \widehat{\boldsymbol{\eta}})$.

For a generic conclusion, we consider the approximation error of $\widetilde{w}(\boldsymbol{\phi}; \widehat{\boldsymbol{\eta}})$ as

$$\left\|\widetilde{w}(\widehat{\boldsymbol{\eta}}) - w^{\{opt\}}\right\|_\infty \leq \left\|\widetilde{w}(\widehat{\boldsymbol{\eta}}) - \widetilde{w}(\boldsymbol{\eta}_0)\right\|_\infty + \left\|\widetilde{w}(\boldsymbol{\eta}_0) - w^{\{opt\}}\right\|_\infty = \epsilon_w + \epsilon_d, \tag{11}$$

where ϵ_d is the specified tolerance in Proposition 2, and ϵ_w is the approximation error of estimating $\widetilde{w}(\boldsymbol{\phi}; \boldsymbol{\eta}_0)$ by $\widetilde{w}(\boldsymbol{\phi}; \widehat{\boldsymbol{\eta}})$. To accommodate a more general distribution on $\widetilde{e}_m$ in addition to Shen et al. (2019)'s work, we consider an alternative perspective on this problem to adopt existing results on non-linear regression (Jennrich, 1969; Wu, 1981). In

the Supplemental Materials Section 3, we show that ϵ_w can be expressed as $\mathcal{O}_p\left(M^{-1/2}\right)$ with four additional regularity conditions. Since M and N are our design parameters, they can be chosen sufficiently large to control the approximation error. One may substitute for it with results from other formulations, for example Theorem 4.1 in Shen et al. (2019) based on sieve estimation.

Algorithm 1 Utilize DNN to estimate $w^{\{opt\}}$

Step 1. Generate training data of size M for DNN. The input data are denoted as $\{\boldsymbol{\phi}_m\}_{m=1}^M$, and the output label is $\{\widehat{w}(\boldsymbol{\phi}_m)\}_{m=1}^M$ in (7).

Step 2. Conduct cross validation to select a proper DNN structure class $\mathcal{F}$ with the training datasets of size $80\% \times M$ and the validation datasets of size $20\% \times M$.

Step 3. Utilize the RMSProp algorithm to train DNN with the selected structure to get $\widehat{\boldsymbol{\eta}}$. Compute $\widetilde{w}(\boldsymbol{\phi};\widehat{\boldsymbol{\eta}})$ as an approximating function of $w^{\{opt\}}(\boldsymbol{\phi})$.

4 Point estimation of θ

In Section 4.1, we illustrate how to construct the ensemble estimator $U(\boldsymbol{x};\widetilde{w})$ following the DNN training in Algorithm 1. Its bias and MSE are further studied at Section 4.2.

4.1 Construct the ensemble estimator $U(\boldsymbol{x};\widetilde{w})$

After obtaining $\widetilde{w}(\boldsymbol{\phi};\widehat{\boldsymbol{\eta}})$ as an estimate of $w^{\{opt\}}(\boldsymbol{\phi})$ from Algorithm 1, we are now ready to construct the ensemble estimator. We denote the variance of T_1 and T_2 in (2) based on $\boldsymbol{x}$

of size n as V_1 and V_2, respectively. Suppose that V_1 and V_2 can be decomposed as follows,

$$V_1 = c_r^{(n)}(\theta, \boldsymbol{\omega}) \times n^{-r}, \tag{12}$$

$$V_2 = c_t^{(n)}(\theta, \boldsymbol{\omega}) \times n^{-t}, \tag{13}$$

where r and t are positive constants, and the leading terms $c_r^{(n)}(\theta, \boldsymbol{\omega})$ and $c_t^{(n)}(\theta, \boldsymbol{\omega})$ are positive as well. For example, if $T_1(\boldsymbol{x})$ is the sample mean of $\boldsymbol{x}$ drawn from a Normal distribution with mean μ and variance σ^2, then $V_1 = \sigma^2/n$ with $c_r^{(n)}(\mu, \sigma) = \sigma^2$ and $r = 1$. Without loss of generality, we assume that $V_1 \leq V_2$, which means that $T_1(\boldsymbol{x})$ is a more precise unbiased estimator as compared with $T_2(\boldsymbol{x})$ at the current sample size n.

Suppose that there exists an unbiased or consistent estimator $\widehat{\boldsymbol{\omega}}(\boldsymbol{x})$ of the nuisance parameters $\boldsymbol{\omega}$. For an observed data vector $\boldsymbol{x}$, we can use $(T_1, \widehat{\boldsymbol{\omega}})$ to estimate $\boldsymbol{\phi} = (\theta, \boldsymbol{\omega})$, and therefore $\widetilde{w}\,(T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}})$ approximates $w^{\{opt\}}(\boldsymbol{\phi})$. Following Algorithm 2, we plug $\widetilde{w}\,(T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}})$ to equation (2), and compute the ensemble estimator of θ as $U\,[\boldsymbol{x}; \widetilde{w}\,(T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}})]$.

Algorithm 2 Point estimate of θ based on observed data $\boldsymbol{x}$

Step 1. Compute the two base estimators T_1 and T_2 of θ, and $\widehat{\boldsymbol{\omega}}$ of $\boldsymbol{\omega}$.

Step 2. Use $\widetilde{w}\,(T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}})$ to estimate $w^{\{opt\}}$.

Step 3. Construct the ensemble estimator of θ as $U\,[\boldsymbol{x}; \widetilde{w}\,(T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}})]$.

4.2 Bias and MSE of $U(\boldsymbol{x}; \widetilde{w})$

For illustrating purposes, we assume that $\widehat{\boldsymbol{\omega}}(\boldsymbol{x})$ is an unbiased estimator of the nuisance parameter $\boldsymbol{\omega}$. The following results can be generalized to cases where $\widehat{\boldsymbol{\omega}}(\boldsymbol{x})$ is consistent. We first introduce two additional conditions before discussing the bias and MSE of the

ensemble estimator,

B.1 The maximum element-wise variance of $\widehat{\boldsymbol{\omega}}(\boldsymbol{x})$ is denoted as V_ω, and it is finite with the following form,

$$V_\omega = \max_{i\in\{1,\cdots,d-1\}} var\left[\widehat{\boldsymbol{\omega}}_i(\boldsymbol{x})\right] = c_s^{(n)}(\theta,\boldsymbol{\omega}) \times n^{-s}, \tag{14}$$

where s and $c_s^{(n)}(\theta,\boldsymbol{\omega})$ are positive.

B.2 The first order partial derivative $\partial\widetilde{w}(\boldsymbol{\phi};\boldsymbol{\eta})/\partial\boldsymbol{\phi}$ and $\partial\widetilde{w}(\boldsymbol{\phi};\boldsymbol{\eta})/\partial\boldsymbol{\eta}$ are upper bounded at $\widetilde{c}_\phi$ and $\widetilde{c}_\eta$, respectively, for $\boldsymbol{\phi}\in\boldsymbol{\Phi}$ and $\boldsymbol{\eta}\in\boldsymbol{H}$.

The notation of V_ω in Condition B.1 is analog to V_1 for T_1 in (12) and V_2 for T_2 in (13). Condition B.2 can be checked empirically based on the fitted DNN $\widetilde{w}(\boldsymbol{\phi};\widehat{\boldsymbol{\eta}})$ obtained in Algorithm 1. Next, we provide upper bounds on the absolute bias and MSE of our estimator $U\left[\boldsymbol{x};\widetilde{w}\left(T_1,\widehat{\boldsymbol{\omega}};\widehat{\boldsymbol{\eta}}\right)\right]$ in the following Theorem 1.

Theorem 1 *Under the aforementioned conditions A.1 - A.4, B.1, B.2, and (11), the absolute bias of* $U(\boldsymbol{x};\widetilde{w}\left[T_1,\widehat{\boldsymbol{\omega}};\widehat{\boldsymbol{\eta}}\right])$ *is upper bounded at,*

$$\begin{aligned}&\left|E\Big\{U(\boldsymbol{x};\widetilde{w}\left[T_1,\widehat{\boldsymbol{\omega}};\widehat{\boldsymbol{\eta}}\right])-\theta\Big\}\right| \leq \\ &\quad |\epsilon_w+\epsilon_d|\times\left(\sqrt{V_1}+\sqrt{V_2}\right)+c_d\times d\times\sqrt{\max(V_1,V_\omega)}\times\left(\sqrt{V_1}+\sqrt{V_2}\right),\end{aligned} \tag{15}$$

where V_1 *is defined in (12),* V_2 *in (13), and* V_ω *in (14). The mean squared error (MSE) of* $U(\boldsymbol{x};\widetilde{w}\left[T_1,\widehat{\boldsymbol{\omega}};\widehat{\boldsymbol{\eta}}\right])$ *is upper bounded at*

$$E\left(\Big\{U(\boldsymbol{x};\widetilde{w}\left[T_1,\widehat{\boldsymbol{\omega}};\widehat{\boldsymbol{\eta}}\right])-\theta\Big\}^2\right) \leq var(\widetilde{U})+S_1+2\sqrt{\Big[S_1+var(\widetilde{U})\Big]\,S_2}+S_2, \tag{16}$$

where

$$\widetilde{U} = w^{\{opt\}}(\boldsymbol{\phi})T_1 + \left[1 - w^{\{opt\}}(\boldsymbol{\phi})\right] T_2, \tag{17}$$

$$S_1 = (c_d)^2 d^2 \max(V_1, V_\omega) var(T_1 - T_2) + 2c_d \times d\sqrt{\max(V_1, V_\omega)}\sqrt{var(T_1 - T_2) var(\widetilde{U})}, \tag{18}$$

$$S_2 = \left[\epsilon_w + \epsilon_d\right]^2 var(T_1 - T_2). \tag{19}$$

We provide some remarks on the upper bound of the absolute bias in (15). The first part can be arbitrarily small by increasing the training data size M and the number of Monte Carlo samples N in Algorithm 1 and choosing a sufficiently small ϵ_d in Proposition 2 as discussed in Section 3.3. We further denote

$$\widetilde{V} = \max(V_1, V_2, V_\omega) = \max\left\{c_r^{(n)}(\theta, \boldsymbol{\omega}) \times n^{-r}, c_t^{(n)}(\theta, \boldsymbol{\omega}) \times n^{-t}, c_s^{(n)}(\theta, \boldsymbol{\omega}) \times n^{-s}\right\}, \tag{20}$$

as the maximum finite variance of T_1, T_2 and $\boldsymbol{\omega}$. The second part shrinks as the sample size n of $\boldsymbol{x}$ increases based on $\widetilde{V}$ in (20). As further demonstrated in the following three experiments, this bias is relatively small in our evaluated finite-sample settings.

On the mean squared error (MSE) of $U(\boldsymbol{x}; \widetilde{w}\,[T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}}])$, we compute its MSE improvement as compared with the component T_1. Note that T_1 has a smaller variance than the

other component T_2 as specified in Section 4.1. The reduction of MSE is

$$
\begin{aligned}
&MSE(T_1) - MSE\left\{U(\boldsymbol{x}; \widetilde{w}\left[T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}}\right])\right\} \\
\geq & var(T_1) - var(\widetilde{U}) - S_1 - \underbrace{\left\{2\sqrt{\left[S_1 + var(\widetilde{U})\right] S_2} + S_2\right\}}_{M_1} \\
\geq & var(T_1 - T_2) \left\{\left(1 - w^{\{opt\}}\right)^2 - \underbrace{(c_d)^2 d^2 \widetilde{V}}_{M_2} - \underbrace{2c_d \times d\sqrt{\widetilde{V}} w^{\{opt\}} \sqrt{\frac{E[T_2(T_1 + T_2)]}{E[T_2(T_2 - T_1)]}}}_{M_3}\right\} - M_1.
\end{aligned}
\tag{21}
$$

The MSE improvement of $U(\boldsymbol{x}; \widetilde{w}\left[T_1, \widehat{\boldsymbol{\omega}}; \widehat{\boldsymbol{\eta}}\right])$ is lowered bounded by (21). The first term $var(T_1 - T_2)\left(1 - w^{\{opt\}}\right)^2$ is non-negative and represents the underlying potential improvement with the unknown $w^{\{opt\}}$ as compared with T_1 based on Proposition 1. For M_1, it can be sufficiently small by our algorithm design following the similar argument of absolute bias on ϵ_w and ϵ_d from S_2 in (19). The next term M_2 decreases as sample size increases per $\tilde{V}$ in (20). The last term M_3 decreases as the covariance between T_1 and T_2 decreases. By Condition A.4 in Section 3.2, the denominator $E[T_2(T_2 - T_1)]$ in M_3 is lower bounded by $c_L/2$. If T_1 and T_2 are highly correlated such that $M_2 + M_3 > \left(1 - w^{\{opt\}}\right)^2$, then there will be no reduction of MSE by using our ensemble estimator. In an extreme scenario where T_1 is the known UMVUE, then $w^{\{opt\}}$ is equal to a constant of 1 with $c_d = 0$. The MSE improvement of our method can be negative, because the lower bound of (21) is $-M_1$.

In practical problems where UMVUE is unknown or does not exist, we suggest applying our proposed method to combine two estimators with a relatively small correlation to obtain a more precise estimator with reduced MSE. If there are more than two candidate unbiased

estimators, we suggest using the two with the smallest empirical variances to get a smaller $\widetilde{V}$ in (20). This construction also ensures that $\widetilde{U}$ with the underlying $w^{\{opt\}}(\boldsymbol{\phi})$ is at least as accurate as either one of them. As a generalization, one can also iteratively apply our algorithm to identify a better statistic with more than two base estimators if necessary.

5 Experiments

In this section, we evaluate the performance of our proposed ensemble estimator in three examples. Section 5.1 considers the scale-uniform distribution, and Section 5.2 assesses a regression model for analyzing heterogeneous data to show our finite sample efficiency gain. In Section 5.3, we apply our analysis method to the Adaptive COVID-19 Treatment Trial (ACTT) to make it more efficient and ethical.

5.1 Scale-uniform family of distributions

We use $Unif$ to denote the Uniform distribution, and consider the scale-uniform distribution $Unif\Big([1-k]\theta, [1+k]\theta\Big)$ with the parameter of interest θ and a known design parameter $k \in (0,1)$ (Galili and Meilijson, 2016). This type of distribution has wide applications, for example the product inventory management in economics (Wanke, 2008) and the inverse transform sampling (Vogel, 2002).

We are interested in making inference on θ using sample $\boldsymbol{x} = (x_1, \cdots, x_n)$ of size n with the support $\Omega_x = \{x \in \mathbb{R} : p_x(x;\theta,k) > 0\}$, where $p_x(x;\theta,k)$ denotes the probability density function of $Unif\Big([1-k]\theta, [1+k]\theta\Big)$. Since the support Ω_x is not the same for all $\theta \in \Theta$ with Θ as an open interval in $\mathbb{R}$, this distribution family does not satisfy the usual differentiability assumptions leading to the Cramér–Rao bound and efficiency of maximum

likelihood estimators (MLEs; Lehmann and Casella (2006), Galili and Meilijson (2016)). We apply the proposed method to construct a more efficient estimator of θ based on existing ones.

As a starting point, we utilize the Rao–Blackwell theorem to construct the first base unbiased estimator T_1. The minimal sufficient statistic for θ is $\left\{x_{(1)}, x_{(n)}\right\}$, where $x_{(1)} = \min(\boldsymbol{x})$ and $x_{(n)} = \max(\boldsymbol{x})$. Since x_1 is unbiased for θ, then an improved unbiased estimator based on the Rao–Blackwell theorem is,

$$\widehat{\theta}_{RB} = E\Big[x_1 | x_{(1)}, x_{(n)}\Big] = \frac{x_{(1)} + x_{(n)}}{2}. \tag{22}$$

The second base estimator T_2 is set as $\widehat{\theta}_M$,

$$\widehat{\theta}_M = \frac{x_{(n)}}{1 + k(n-1)/(n+1)}, \tag{23}$$

which is the unbiased corrected version of the MLE $\widehat{\theta}_{MLE} = x_{(n)}/(k+1)$ (Galili and Meilijson, 2016),

Utilizing our proposed method, we ensemble $T_1 = \widehat{\theta}_{RB}$ and $T_2 = \widehat{\theta}_M$ in (2) to construct a better estimator $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_M\right)$ with a smaller variance. Suppose we are interested in $\theta \in \Theta = (0.2, 10)$ as an open interval in $\mathbb{R}$ with finite data size $n = 2$ or 10. For a given n, we simulate $M = 10^3$ training input data for DNN with varying $\theta \sim Unif(0.2, 10)$ and the known parameter k at either 0.1 or 0.9 to accommodate the scenarios considered at Table 1 for evaluating performance. Note that the above training data sample spaces can be set wider as needed. The input data of DNN is $\boldsymbol{\phi} = (\theta, k)$, and the output label $\widehat{w}(\boldsymbol{\phi})$ in (7) is evaluated by $N = 10^6$ Monte Carlo samples. In cross-validation, we consider 4 candidate DNN structures: 2 hidden layers with 40 nodes per layer, 2 hidden layers with

60 nodes per layer, 3 hidden layers with 40 nodes per layer, 3 hidden layers with 60 nodes per layer, and select the structure with the smallest validation MSE for final DNN training. We use a dropout rate of 0.1, number of training epochs at 10^3, and a batch size of 100 in the training process to obtain a fitted DNN $\widetilde{w}(\boldsymbol{\phi}; \widehat{\boldsymbol{\eta}})$. The number of simulation iterations for testing at Table 1 is 10^6. The above setup parameters of training DNN are utilized throughout this article if not specified otherwise.

Under all scenarios considered in Table 1, the relative bias of $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_M\right)$ is less than 10^{-3} (results not shown). To further evaluate the efficiency gain of our method, we compute the relative efficiency of $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_M\right)$ versus three existing estimators: $\widehat{\theta}_{RB}$ in (22), $\widehat{\theta}_M$ in (23) and $\widehat{\theta}_E$ as the sample mean. The relative efficiency of two estimators is defined as the inverse ratio of their variances. The ensemble estimator $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_M\right)$ is uniformly more efficient than three comparators as demonstrated by all ratios larger than 1. Within the same n, one observes that $\widehat{\theta}_{RB}$ is more efficient than $\widehat{\theta}_M$ when $k = 0.1$, and vice versa when $k = 0.9$. Our $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_M\right)$ learns their advantages under different k's and shows a consistently better performance.

5.2 Regression model for analyzing heterogeneous data

Aggregating and analyzing heterogeneous data is one of the most fundamental challenges in scientific data analysis (Fan et al., 2018). For observable $\boldsymbol{X} \in \mathbb{R}^d$ and a discrete variable $Z \in \mathcal{Z}$, a general mixture model assumes,

$$\boldsymbol{X}|(Z = z) \sim \mathcal{F}(\boldsymbol{\theta}_z),$$

				Relative efficiency versus		
n	k	θ	SD	$\widehat{\theta}_{RB}$	$\widehat{\theta}_{M}$	$\widehat{\theta}_{E}$
2	0.1	0.5	0.020	1.003	1.254	1.003
		1	0.041	1.003	1.253	1.003
		5	0.204	1.003	1.252	1.003
	0.9	0.5	0.163	1.271	1.002	1.271
		1	0.326	1.268	1.002	1.268
		5	1.632	1.270	1.002	1.270
10	0.1	0.5	0.006	1.008	1.566	2.220
		1	0.012	1.009	1.564	2.221
		5	0.061	1.008	1.570	2.217
	0.9	0.5	0.043	1.665	1.003	3.669
		1	0.086	1.665	1.003	3.665
		5	0.430	1.658	1.003	3.653

Table 1: Standard deviation (SD) of $U\left(\widehat{\theta}_{RB}, \widehat{\theta}_{M}\right)$ and its high relative efficiency versus two base components $\widehat{\theta}_{RB}$ and $\widehat{\theta}_{M}$, and the empirical mean $\widehat{\theta}_{E}$.

for a distribution $\mathcal{F}$ with parameters $\boldsymbol{\theta}_z$ in the sub-population z (Fan et al., 2018). The variable Z can be known in some applications, for example on synthesizing control information from multiple historical clinical trials (Neuenschwander et al., 2010); or it can be latent in general (Fan et al., 2014).

In this motivating simulation study, we consider the following Gaussian regression model where the variance of the dependent variable is proportional to the square of its expected value (Amemiya, 1973; Ramanathan, 2002),

$$y_i \sim \mathcal{N}\left(\boldsymbol{x}_i'\boldsymbol{\theta}, \left[\boldsymbol{x}_i'\boldsymbol{\theta}\right]^2\right),$$

where $\boldsymbol{x}_i$ is a vector of covariates for subject i, and $\boldsymbol{\theta}$ is a vector of unknown parameters. This type of model has wide applications in economics and operational research, for example understanding the influence of customer demographics on the rent paid (Anderson and Jaggia, 2009), and modeling efficiency scores in censoring data generating process (McDonald, 2009).

Challenges exist in this problem to find an efficient unbiased estimator of $\boldsymbol{\theta}$ in finite samples. The minimal sufficient statistics consisting of sample mean and sample variance are not complete for $\boldsymbol{\theta} \in \Theta$ (Khan, 2015). When $\boldsymbol{x}_i'\boldsymbol{\theta}$ is relatively small, the Fisher information matrix can be ill-conditioned (Amemiya, 1973), which introduces bias in the maximum likelihood estimator (MLE). As robust alternatives, Amemiya (1973) considers the following two unbiased estimators,

$$\widehat{\boldsymbol{\theta}}_L = \left[\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i'\right]^{-1} \sum_{i=1}^{n} \boldsymbol{x}_i y_i \tag{24}$$

$$\widehat{\boldsymbol{\theta}}_W = \left[\sum_{i=1}^{n} \frac{1}{\left(\boldsymbol{x}_i'\widehat{\boldsymbol{\theta}}_L\right)^2} \boldsymbol{x}_i \boldsymbol{x}_i'\right]^{-1} \sum_{i=1}^{n} \frac{1}{\left(\boldsymbol{x}_i'\widehat{\boldsymbol{\theta}}_L\right)^2} \boldsymbol{x}_i y_i, \tag{25}$$

where $\widehat{\boldsymbol{\theta}}_L$ is the least square estimator and $\widehat{\boldsymbol{\theta}}_W$ is the weighted least square estimators. To avoid extreme values in practice, we upper bound the weight $1/\left(\boldsymbol{x}_i'\widehat{\boldsymbol{\theta}}_L\right)^2$ by 10^5. Matrix $\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i'$ is assumed to be positive definite and $\boldsymbol{x}_i$ is bounded for $i = 1, \cdots, n$. We utilize our proposed method to assemble $\widehat{\boldsymbol{\theta}}_W$ as $\boldsymbol{T}_1$ and $\widehat{\boldsymbol{\theta}}_L$ as $\boldsymbol{T}_2$ to get a DNN-based estimator $\boldsymbol{U}\left(\widehat{\boldsymbol{\theta}}_W, \widehat{\boldsymbol{\theta}}_L\right)$ with smaller variance.

In this simulation study, we consider that $\boldsymbol{\theta}$ is a four-dimensional vector with θ_1 as intercept and θ_2, θ_3 and θ_4 as coefficients. Further denote U_1, U_2, U_3 and U_4 as the

elements in our ensemble estimator: $\boldsymbol{U}\left(\widehat{\boldsymbol{\theta}}_W, \widehat{\boldsymbol{\theta}}_L\right) = (U_1, U_2, U_3, U_4)$. The parameter space for θ_1, θ_2, θ_3 and θ_4 are considered at $\Theta = (-1.5, 1.5)$. Covariates $\boldsymbol{x}_i$, for $i = 1, \cdots, n$, is simulated from a uniform distribution with a lower bound -2 and an upper bound 2. A moderate sample size $n = 100$ is evaluated in this study. In Algorithm 2, we simulate $M = 10^3$ training input data for DNN as $\boldsymbol{\phi} = (\theta_1, \theta_2, \theta_3, \theta_4)$ with varying θ_1, θ_2, θ_3 and θ_4 from uniform distributions within Θ. The number of Monte Carlo samples is $N = 10^5$ when computing $\widehat{w}(\boldsymbol{\phi})$ in (7). In the testing stage, we evaluate different patterns of $\boldsymbol{\theta}$ with three magnitudes at 0.2, 0.6 and 1.2 at Table 2.

The absolute relative bias of our estimator is less than 0.02 across all scenarios. Table 2 shows the finite sample efficiency, where $\widehat{\boldsymbol{\theta}}_W$ is generally more efficient than $\widehat{\boldsymbol{\theta}}_L$ and can be less efficient on estimating θ_4 in some cases. Our proposed estimator $\boldsymbol{U}\left(\widehat{\boldsymbol{\theta}}_W, \widehat{\boldsymbol{\theta}}_L\right)$ is consistently more efficient than its two components $\widehat{\boldsymbol{\theta}}_W$ and $\widehat{\boldsymbol{\theta}}_L$ under all scenarios (relative efficiencies are larger than one).

5.3 Adaptive COVID-19 Treatment Trial (ACTT)

In this section, we apply our method to the Adaptive COVID-19 Treatment Trial (ACTT) to evaluate the safety and efficacy of remdesivir from Gilead Inc. in hospitalized adults diagnosed with COVID-19 (National Institutes of Health, 2020a). Adaptive clinical trials are appealing under the COVID-19 pandemic with limited knowledge on treatment profiles under evaluation, because they are capable of accommodating uncertainty during study conduction. As acknowledged by regulatory agencies (Food and Drug Administration, 2019; European Medicines Agency, 2007), the bias correction in adaptive design is still a less well-studied phenomenon. Our proposed method not only provides a solution for this problem to have an accurate understanding of the treatment effect, but also improves finite

				Relative efficiency versus $\widehat{\boldsymbol{\theta}}_W$				Relative efficiency versus $\widehat{\boldsymbol{\theta}}_L$			
θ_1	θ_2	θ_3	θ_4	U_1	U_2	U_3	U_4	U_1	U_2	U_3	U_4
0.2	0.2	0.2	0.2	1.186	1.301	1.388	1.394	1.961	1.605	1.790	1.575
0.2	0.2	0.2	-0.2	1.336	1.372	1.479	1.536	1.755	1.572	1.586	1.475
0.2	-0.2	-0.2	-0.2	1.582	1.598	1.481	1.617	1.605	1.427	1.684	1.395
-0.2	-0.2	-0.2	-0.2	1.162	1.306	1.392	1.390	1.921	1.617	1.789	1.565
0.6	0.6	0.6	0.6	1.321	1.384	1.465	1.471	1.947	1.525	1.660	1.472
0.6	0.6	0.6	-0.6	1.445	1.420	1.546	1.612	1.729	1.498	1.486	1.399
0.6	-0.6	-0.6	-0.6	1.616	1.671	1.518	1.660	1.535	1.381	1.612	1.374
-0.6	-0.6	-0.6	-0.6	1.297	1.380	1.467	1.469	1.914	1.527	1.674	1.473
1.2	1.2	1.2	1.2	1.330	1.409	1.491	1.508	1.929	1.519	1.653	1.468
1.2	1.2	1.2	-1.2	1.437	1.430	1.593	1.629	1.689	1.476	1.485	1.386
1.2	-1.2	-1.2	-1.2	1.646	1.690	1.537	1.637	1.554	1.388	1.623	1.351
-1.2	-1.2	-1.2	-1.2	1.338	1.414	1.492	1.499	1.928	1.527	1.655	1.465

Table 2: High relative efficiency of $\boldsymbol{U}\left(\widehat{\boldsymbol{\theta}}_W, \widehat{\boldsymbol{\theta}}_L\right)$ versus two base components $\widehat{\boldsymbol{\theta}}_W$ and $\widehat{\boldsymbol{\theta}}_L$.

sample efficiency of such estimators to make adaptive designs more efficient and ethical.

For illustrative purposes, we consider the sample size reassessment adaptive design with a binary endpoint of achieving hospital discharge at Day 14 (National Institutes of Health, 2020b; Gilead Inc., 2020). Let θ_1 be the response rate in the placebo, and θ_2 be that from the treatment. The objective is to estimate the treatment effect $\theta = \theta_2 - \theta_1$ based on binary data from two groups. The underlying true $\theta_1 = 0.47$ and $\theta_2 = 0.59$ are assumed based on the preliminary interim results in National Institutes of Health (2020b).

We consider a two-stage adaptive design, where $n^{(1)}$ subjects are randomized to the treatment group and $n^{(1)}$ subjects to the control group in the first stage. After evaluating unblinded interim data from those $2 \times n^{(1)}$ subjects, a Data and Safety Monitoring Board

(DSMB) makes sample size adjustments based on the following rule,

$$n^{(2)} = \begin{cases} n^{(2)}_{min}, & \text{if } \widehat{\theta}\left\{\boldsymbol{x}_2^{(1)}\right\} - \widehat{\theta}\left\{\boldsymbol{x}_1^{(1)}\right\} > \theta_{min} \\ n^{(2)}_{max}, & \text{otherwise} \end{cases} \tag{26}$$

where $\widehat{\theta}\left\{\boldsymbol{x}_j^{(h)}\right\}$ is the sample average, $\boldsymbol{x}_j^{(h)}$ is a vector of observed binary data of size $n^{(h)}$ for group j, $j = 1, 2$ at stage h, $h = 1, 2$, and $n^{(2)}_{min}$, $n^{(2)}_{max}$ and θ_{min} are pre-specified design features. Basically, $n^{(2)}$ in the second stage will be decreased to $n^{(2)}_{min}$ if a promising treatment effect larger than a clinically meaningful difference θ_{min} is observed, but increased to $n^{(2)}_{max}$ otherwise. Other adaptive rules can also be applied (Bretz et al., 2009). Due to the pre-specified adjustment of $n^{(2)}$ based on the first stage data, it is challenging to determine the existence or to characterize the functional form of the complete sufficient statistics of θ. The empirical treatment difference $\widehat{\theta}(\boldsymbol{x}_2) - \widehat{\theta}(\boldsymbol{x}_1)$ is even a biased estimator of θ (Bretz et al., 2009), where $\boldsymbol{x}_j = \left\{\boldsymbol{x}_j^{(1)}, \boldsymbol{x}_j^{(2)}\right\}$ is the pooled data from two stages in group j, $j = 1, 2$.

An unbiased estimator of θ can be constructed by the following weighted average of the treatment differences from two stages based on the conditional invariance principle (Bretz et al., 2009),

$$\widetilde{\theta}(k) = k\Delta^{(1)} + (1-k)\Delta^{(2)}, \tag{27}$$

where $k \in [0, 1]$ is a constant, and $\Delta^{(h)} = \widehat{\theta}\left\{\boldsymbol{x}_2^{(h)}\right\} - \widehat{\theta}\left\{\boldsymbol{x}_1^{(h)}\right\}$ is an unbiased estimator of θ based on data at stage h, for $h = 1, 2$. The pre-specified weight k can be chosen to minimize the variance of $\widetilde{\theta}(k)$ in the study design stage given a working value of the true treatment effect θ, but may lead to efficiency loss when observed data deviate. Using our proposed method, we ensemble $T_1 = \widetilde{\theta}(0.5)$ and $T_2 = \Delta^{(1)}$ to deliver a more accurate unbiased estimator within a neighborhood of the underlying θ.

We consider $\theta_1 \in (0.2, 0.7)$ and $\theta \in (-0.2, 0.3)$ as our parameter spaces, and $n^{(1)} = 100$, $n^{(2)}_{min} = 50$, $n^{(2)}_{max} = 250$ and $\theta_{min} = 0.16$ as design features in (26). Following Algorithm 2, we simulate $M = 10^3$ training input data for DNN with varying θ and θ_1 from uniform distributions within their corresponding supports. The input data vector for DNN is $\boldsymbol{\phi} = (\theta_1, \theta)$. The performance of our DNN based estimator $U\left\{\widetilde{\theta}(0.5), \Delta^{(1)}\right\}$ is compared with three unbiased estimators $\widetilde{\theta}(0.2)$, $\widetilde{\theta}(0.5)$ and $\widetilde{\theta}(0.8)$ in (27) with $k = 0.2, 0.5$, and 0.8, respectively.

In the first block of Table 3, these 4 scenarios cover varying magnitudes of θ_1 around its true value 0.47, and with $\theta_2 = \theta_1$ demonstrating no treatment effect. The next three blocks consider varying placebo rate θ_1 and varying treatment effect θ. Under all scenarios evaluated, our ensemble estimator has a relatively small bias ≤ 0.001. Among the three comparators, $\widehat{\theta}(0.2)$ is more accurate when $\theta = 0$, and $\widehat{\theta}(0.5)$ is preferable when $\theta > 0$. Our estimator is consistently more efficient than them, supported by the relative efficiency.

We then plot the power of rejecting the one-sided null hypothesis $H_0 : \theta \leq 0$ at a type I error rate $\alpha = 0.05$ under $\theta_1 = 0.47$ and varying treatment effect θ in Figure 2. The critical values of rejecting H_0 are computed at 0.064 for our method, 0.064 for $\widetilde{\theta}(0.2)$, 0.068 for $\widetilde{\theta}(0.5)$, and 0.094 for $\widetilde{\theta}(0.8)$ by the grid search method to control validating type I error rates not exceeding 5% when $\theta_1 = \theta_2 = 0.42, 0.5, 0.58, 0.66$. Our proposed method has consistently higher power of detecting a promising treatment effect than the other three estimators. Therefore, a more efficient and more ethical adaptive clinical trial can be implemented based on our proposed method to evaluate treatment options to cure COVID-19.

			$U\left\{\widetilde{\theta}(0.5), \Delta^{(1)}\right\}$		Relative efficiency versus		
θ_1	θ_2	θ	Bias	SD	$\widetilde{\theta}(0.2)$	$\widetilde{\theta}(0.5)$	$\widetilde{\theta}(0.8)$
0.42	0.42	0	0.001	0.039	1.012	1.165	2.157
0.50	0.50		0.001	0.039	1.009	1.162	2.150
0.58	0.58		0.001	0.039	1.010	1.166	2.160
0.66	0.66		0.001	0.037	1.011	1.173	2.178
0.42	0.52	0.1	0.001	0.045	1.203	1.034	1.607
	0.54	0.12	0.001	0.047	1.292	1.019	1.477
	0.56	0.14	< 0.001	0.049	1.386	1.010	1.361
0.47	0.57	0.1	0.001	0.045	1.197	1.037	1.625
	0.59	0.12	0.001	0.047	1.289	1.021	1.487
	0.61	0.14	< 0.001	0.049	1.385	1.009	1.358
0.52	0.62	0.1	0.001	0.045	1.205	1.037	1.612
	0.64	0.12	0.001	0.047	1.296	1.017	1.473
	0.66	0.14	< 0.001	0.049	1.392	1.009	1.354

Table 3: Small bias of $U\left\{\widetilde{\theta}(0.5), \Delta^{(1)}\right\}$ and its high relative efficiency compared with three unbiased estimators in the ACTT on COVID-19.

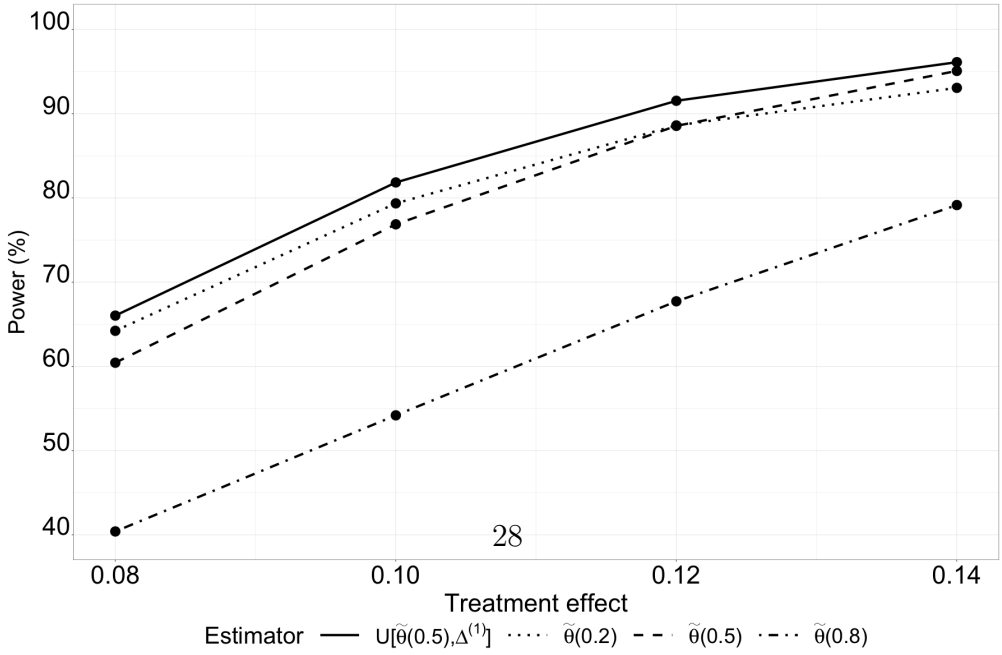

Figure 2: Consistently higher power of $U\left\{\widetilde{\theta}(0.5), \Delta^{(1)}\right\}$ than $\widehat{\theta}(0.2)$, $\widehat{\theta}(0.5)$ and $\widehat{\theta}(0.8)$ to detect a promising treatment effect θ in the ACTT on COVID-19.

6 Discussion

In this article, we propose a novel DNN-based ensemble learning method to improve finite sample efficiency of point estimation. As a critical application in the ACTT on COVID-19, our method is more efficient and has a higher power of detecting a promising treatment effect than several alternatives. The proposed method can contribute to a more ethical and efficient adaptive clinical trial with fewer patients enrolled.

Our construction in (2) is to get the best linear unbiased estimator when the optimal weight is known. In practice when the weight is to be estimated from data, we show that the bias approaches zero as sample size increases. This construction on correcting bias is preferred in many applications, for example in understanding the treatment effect of the study drug relative to placebo in the ACTT on COVID-19. Our method can be generalized to minimize other measures such as MSE, Bayes risk, et cetera. For instance, the Ridge estimator can be combined to reduce MSE by introducing a tolerable bias. One can also iteratively apply our method to integrate more than two base estimators.

There are some potential limitations of our method. The DNN-based approach requires additional training and computational time to obtain the estimator. It takes approximately 4 hours to simulate training and validation data to reproduce Table 3 in the case study. However, the well-trained DNNs can be saved in files before observing current data. As illustrated in our shared code, one can instantly compute the weight parameter and construct the ensemble estimator with available functional form of DNNs. A future work is to make statistical inference of the parameters of interest based on the ensemble estimator.

Supplementary Materials

Supplementary Materials are available online including the R code and a help file to replicate all simulation studies.

Acknowledgments

This manuscript was supported by AbbVie. AbbVie participated in the review and approval of the content. Tianyu Zhan is employed by AbbVie Inc., Haoda Fu is employed by Eli Lilly and Company, and Jian Kang is Professor in the Department of Biostatistics at the University of Michigan, Ann Arbor. All authors may own AbbVie stock.

References

Amemiya, T. (1973). Regression analysis when the variance of the dependent variable is proportional to the square of its expectation. *Journal of the American Statistical Association 68*(344), 928–934.

Anderson, M. H. and S. Jaggia (2009). Rent-to-own agreements: Customer characteristics and contract outcomes. *Journal of Economics and Business 61*(1), 51–69.

Anthony, M. and P. L. Bartlett (2009). *Neural network learning: Theoretical foundations*. Cambridge University Press.

Bach, F. (2017). Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research 18*(1), 629–681.

Bai, J., Q. Song, and G. Cheng (2020). Efficient variational inference for sparse deep learning with theoretical guarantee. *arXiv preprint arXiv:2011.07439*.

Biau, G. (2012). Analysis of a random forests model. *The Journal of Machine Learning Research 13*(1), 1063–1095.

Biau, G., E. Scornet, and J. Welbl (2019). Neural random forests. *Sankhya A 81*(2), 347–386.

Bradic, J. et al. (2016). Randomized maximum-contrast selection: Subagging for large-scale regression. *Electronic Journal of Statistics 10*(1), 121–170.

Brahma, P. P., D. Wu, and Y. She (2015). Why deep learning works: A manifold disentanglement perspective. *IEEE transactions on neural networks and learning systems 27*(10), 1997–2008.

Bretz, F., F. Koenig, W. Brannath, E. Glimm, and M. Posch (2009). Adaptive designs for confirmatory clinical trials. *Statistics in Medicine 28*(8), 1181–1217.

Chao, S.-K., Z. Wang, Y. Xing, and G. Cheng (2020). Directional pruning of deep neural networks. *arXiv preprint arXiv:2006.09358*.

Chen, H., Z. Mo, Z. Yang, and X. Wang (2019). Theoretical investigation of generalization bound for residual networks. *IJCAI*, 2081–2087.

Chen, M.-H., D. K. Dey, P. Müller, D. Sun, and K. Ye (2010). Bayesian clinical trials. *Frontiers of Statistical Decision Making and Bayesian Analysis*, 257–284.

Chen, M.-H., J. G. Ibrahim, D. Zeng, K. Hu, and C. Jia (2014). Bayesian design of superiority clinical trials for recurrent events data with applications to bleeding and transfusion events in myelodyplastic syndrome. *Biometrics 70*(4), 1003–1013.

Chen, T. and C. Guestrin (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Chen, Y., Q. Gao, F. Liang, and X. Wang (2020). Nonlinear variable selection via deep neural networks. *Journal of Computational and Graphical Statistics*, 1–9.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems 2*(4), 303–314.

European Medicines Agency (2007). Reflection paper on methodological issues in confirmatory clinical trials planned with an adaptive design. *EMEA*.

Evans, L. C. (2010). *Partial differential equations*. American Mathematical Society.

Fan, J., F. Han, and H. Liu (2014). Challenges of big data analysis. *National Science Review 1*(2), 293–314.

Fan, J., H. Liu, W. Wang, and Z. Zhu (2018). Heterogeneity adjustment with applications to graphical model inference. *Electronic Journal of Statistics 12*(2), 3908.

Fan, J., H. Liu, Z. Wang, and Z. Yang (2018). Curse of heterogeneity: Computational barriers in sparse mixture models and phase retrieval. *arXiv preprint arXiv:1808.06996*.

Food and Drug Administration (2019). Adaptive design clinical trials for drugs and biologics guidance for industry. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/adaptive-design-clinical-trials-drugs-and-biologics-guidance-industry.

Galili, T. and I. Meilijson (2016). An example of an improvable Rao-Blackwell improvement, inefficient maximum likelihood estimator, and unbiased generalized bayes estimator. *The American Statistician 70*(1), 108–113.

Gao, Q. and X. Wang (2021). Theoretical investigation of generalization bounds for adversarial learning of deep neural networks. *Journal of Statistical Theory and Practice 15*(2), 1–28.

Gilead Inc. (2020). Gilead Announces Results From Phase 3 Trial of Investigational Antiviral Remdesivir in Patients With Severe COVID-19. https://www.gilead.com/news-and-press/press-room/press-releases/2020/4/gilead-announces-results-from-phase-3-trial-of-investigational-antiviral-remdesivir-in-patients-with-severe-covid-19.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning*. MIT press.

Hinton, G., N. Srivastava, and K. Swersky (2012). Neural networks for machine learning. *Coursera, video lectures 307*.

Jennrich, R. I. (1969). Asymptotic properties of non-linear least squares estimators. *The Annals of Mathematical Statistics 40*(2), 633–643.

Katzfuss, M., J. R. Stroud, and C. K. Wikle (2016). Understanding the ensemble kalman filter. *The American Statistician 70*(4), 350–357.

Khan, R. A. (2015). A remark on estimating the mean of a normal distribution with known coefficient of variation. *Statistics 49*(3), 705–710.

Lehmann, E. L. and G. Casella (2006). *Theory of point estimation*. Springer Science & Business Media.

Liang, S., W. Lu, and R. Song (2018). Deep advantage learning for optimal dynamic treatment regime. *Statistical theory and related fields 2*(1), 80–88.

Lu, Y. Y., Y. Fan, J. Lv, and W. S. Noble (2018). Deeppink: reproducible feature selection in deep neural networks. *Advances in Neural Information Processing Systems*.

McDermott, P. L. and C. K. Wikle (2017). An ensemble quadratic echo state network for non-linear spatio-temporal forecasting. *Stat 6*(1), 315–330.

McDonald, J. (2009). Using least squares and tobit in second stage DEA efficiency analyses. *European Journal of Operational Research 197*(2), 792–798.

National Institutes of Health (2020a). Adaptive COVID-19 Treatment Trial (ACTT). https://clinicaltrials.gov/ct2/show/NCT04280705.

National Institutes of Health (2020b). NIH Clinical Trial Shows Remdesivir Accelerates Recovery from Advanced COVID-19. https://www.niaid.nih.gov/news-events/nih-clinical-trial-shows-remdesivir-accelerates-recovery-advanced-covid-19.

Neuenschwander, B., G. Capkun-Niggli, M. Branson, and D. J. Spiegelhalter (2010). Summarizing historical information on controls in clinical trials. *Clinical Trials 7*(1), 5–18.

Ramanathan, R. (2002). *Introductory econometrics with applications*. Harcount College Publishers.

Rava, D. and J. Bradic (2020). Deephazard: neural network for time-varying risks. *arXiv preprint arXiv:2007.13218*.

She, Y., Y. He, and D. Wu (2014). Learning topology and dynamics of large recurrent neural networks. *IEEE Transactions on Signal Processing 62*(22), 5881–5891.

Shen, J. and H. Yu (2010). Efficient spectral sparse grid methods and applications to high-dimensional elliptic problems. *SIAM Journal on Scientific Computing 32*(6), 3228–3250.

Shen, L. (2001). An improved method of evaluating drug effect in a multiple dose clinical trial. *Statistics in Medicine 20*(13), 1913–1929.

Shen, X., C. Jiang, L. Sakhanenko, and Q. Lu (2019). Asymptotic Properties of Neural Network Sieve Estimators. *arXiv preprint arXiv:1906.00875*.

Stallard, N., S. Todd, and J. Whitehead (2008). Estimation following selection of the largest of two normal means. *Journal of Statistical Planning and Inference 138*(6), 1629–1638.

Tian, Y. and Y. Feng (2021). Rase: Random subspace ensemble classification. *Journal of Machine Learning Research 22*(45), 1–93.

Vogel, C. R. (2002). *Computational methods for inverse problems*, Volume 23. Siam.

Wang, D., H. Fu, and P.-L. Loh (2020). Boosting algorithms for estimating optimal individualized treatment rules. *arXiv preprint arXiv:2002.00079*.

Wanke, P. F. (2008). The uniform distribution as a first practical approach to new product inventory management. *International Journal of Production Economics 114*(2), 811–819.

White, H. (1990). Connectionist nonparametric regression: Multilayer feedforward networks can learn arbitrary mappings. *Neural Networks 3*(5), 535–549.

Wu, C.-F. (1981). Asymptotic theory of nonlinear least squares estimation. *The Annals of Statistics*, 501–513.

Wu, H., Y. Fan, and J. Lv (2020). Statistical insights into deep neural network learning in subspace classification. *Stat 9*(1), e273.

Xu, Y. and X. Wang (2018). Understanding weight normalized deep neural networks with rectified linear units. *Advances in Neural Information Processing Systems*, 130–139.

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

Yuan, Y., Y. Deng, Y. Zhang, and A. Qu (2020). Deep learning from a statistical perspective. *Stat 9*(1), e294.

Zhang, G., C. Webster, M. Gunzburger, and J. Burkardt (2015). A hyperspherical adaptive sparse-grid method for high-dimensional discontinuity detection. *SIAM Journal on Numerical Analysis 53*(3), 1508–1536.