Forthcoming in Econometrica

DEEP NEURAL NETWORKS FOR ESTIMATION AND INFERENCE[1]

Max H. Farrell, Tengyuan Liang[2] and and Sanjog Misra[3]

We study deep neural networks and their use in semiparametric inference. We establish novel nonasymptotic high probability bounds for deep feedforward neural nets. These deliver rates of convergence that are sufficiently fast (in some cases minimax optimal) to allow us to establish valid second-step inference after first-step estimation with deep learning, a result also new to the literature. Our nonasymptotic high probability bounds, and the subsequent semiparametric inference, treat the current standard architecture: fully connected feedforward neural networks (multilayer perceptrons), with the now-common rectified linear unit activation function, unbounded weights, and a depth explicitly diverging with the sample size. We discuss other architectures as well, including fixed-width, very deep networks. We establish the nonasymptotic bounds for these deep nets for a general class of nonparametric regression-type loss functions, which includes as special cases least squares, logistic regression, and other generalized linear models. We then apply our theory to develop semiparametric inference, focusing on causal parameters for concreteness, and demonstrate the effectiveness of deep learning with an empirical application to direct mail marketing.

Keywords: Deep Neural Networks, Rectified Linear Unit, Nonasymptotic Bounds, Convergence Rates, Semiparametric Inference, Treatment Effects, Program Evaluation.

1. Introduction

Statistical machine learning methods are being rapidly integrated into the social and medical sciences. Economics is no exception, and there has been a recent surge of research that applies and explores machine learning methods in the context of econometric modeling, particularly in "big data" settings. Furthermore, theoretical properties of these methods are the subject of intense recent study. Our goal in the present work is to study a particular statistical machine learning technique which is widely popular in industrial applications, but less frequently used in academic work and largely ignored in recent theoretical developments on inference: deep neural networks.

Neural networks are estimation methods that model the relationship between inputs and outputs using layers of connected computational units (neurons), patterned after the biological neural networks of brains. These computational units sit between the inputs and output and allow data-driven learning of the appropriate model, in addition to learning the parameters of that model. Put into terms more familiar in nonparametric econometrics: neural networks can be thought of as a (complex) type of linear sieve estimation where the basis functions themselves are flexibly learned from the data by optimizing over many combinations of simple functions. Neural networks are perhaps not as familiar to economists as other methods, and indeed, were out of favor in the machine learning community for several years, returning to prominence only very recently in the form of deep learning. Deep neural nets contain many hidden layers of neurons between the input and output layers, and have been found to exhibit superior performance across a variety of contexts. Our work aims to bring wider attention to these methods and to fill some gaps in the theoretical understanding of inference using deep neural networks.

Before the recent surge in attention, neural networks had taken a back seat to other methods (such as kernel methods or forests) largely because of their modest empirical performance and challenging optimization. Before falling out of favor, neural networks were widely studied and applied, particularly in the 1990s. In that time, *shallow* neural networks with *smooth* activation functions were shown to have many good theoretical properties (Anthony and Bartlett, 1999; Chen and White, 1999; White, 1992). However, the availability of

University of Chicago, Booth School of Business
max.farrell@chicagobooth.edu; tengyuan.liang@chicagobooth.edu; sanjog.misra@chicagobooth.edu

[1] We thank Milica Popovic for outstanding research assistance. We thank Guido Imbens, the co-editor, and two anonymous reviewers, as well as Alex Belloni, Xiaohong Chen, Denis Chetverikov, Chris Hansen, Whitney Newey, and Andres Santos, for thoughtful comments, suggestions, and discussions that substantially improved the paper.

[2] Liang gratefully acknowledges support from the George C. Tiao Fellowship.

[3] Misra gratefully acknowledges support from the Neubauer Family Foundation.

scalable computing and stochastic optimization techniques (Kingma and Ba, 2014; LeCun et al., 1998) and the changes from shallow to deep networks and from smooth sigmoid-type activation functions to rectified linear units (ReLU), $x \mapsto \max(x, 0)$ (Nair and Hinton, 2010), have seemingly overcome optimization hurdles and empirical issues, and now this form of deep learning matches or sets the state of the art in many prediction contexts. Our theoretical results speak directly to this modern implementation of deep learning: we focus on the ReLU activation function, explicitly model the depth of the network as diverging with the sample size, and do not require bounded weights.

We provide nonasymptotic high probability bounds for nonparametric estimation using deep neural networks for a large class of statistical models. Our bounds appear to be new to the literature and are the main theoretical contributions of the paper. We provide results for a general class of smooth loss functions for nonparametric regression style problems, covering as special cases generalized linear models and other empirically useful contexts. For example, in our application to causal inference we specialize our results to linear and logistic regression as concrete illustrations. Our bounds immediately yield empirical and population L_2 convergence rates. For a certain architecture we obtain the optimal rate. Our proof strategy employs a localization analysis that uses scale-insensitive measures of complexity, allowing us to consider richer classes of neural networks. This is in contrast to analyses which restrict the networks to have bounded parameters for each unit (discussed more below) and to the application of scale sensitive measures such as metric entropy. These approaches would not deliver our sharp bounds and fast rates when treating standard, feasible neural networks. Recent developments in approximation theory and complexity for deep ReLU networks are important building blocks for our results.

We follow our main results by applying our nonasymptotic high probability bounds to deliver valid inference on finite-dimensional parameters following first-step estimation using deep learning. Our aim is not to innovate at the semiparametric step but to utilizing existing results. Our work contributes directly to this area of research by showing that deep nets are a valid and useful first-step estimator for semiparametric inference in general. Further, we show that inference after deep learning may not require sample splitting or cross fitting. In particular, we use localization to directly verify conditions required for valid inference, which may be a novel application of this proof method that is useful in future problems of inference following machine learning.

We illustrate these ideas in the context of causal inference for concreteness and wide applicability, as well as to allow direct comparison to the literature. Program evaluation with observational data is one of the most common and important inference problems, and has often been used as a test case for theoretical study of inference following machine learning (e.g., Athey et al., 2018; Belloni et al., 2017, 2014; Farrell, 2015). Deep neural networks have been argued (experimentally) to outperform the previous state-of-the-art (Hartford et al., 2017; Shalit et al., 2017; Westreich et al., 2010). We establish valid inference for treatment effects and counterfactual expected utility/profits from treatment targeting strategies. We note that the selection on observables framework yields identification of counterfactual average outcomes without additional structural assumptions, so that, e.g., expected profit from a counterfactual treatment rule can be evaluated.

We numerically illustrate our results, and more generally the utility of deep learning, with an empirical study of a direct mail marketing campaign. Our data come from a large US consumer products retailer and consists of around three hundred thousand consumers with one hundred fifty covariates. Hitsch and Misra (2018) recently used this data to study various estimators, both traditional and modern, of heterogeneous treatment effects. We study the effect of catalog mailings on consumer purchases, and moreover, compare different targeting strategies (i.e. to which consumers catalogs should be mailed). The cost of sending out a single catalog can be close to one dollar, and with millions being set out, carefully assessing the targeting strategy is crucial. Our results suggest that deep nets are at least as good as (and sometimes better) that the best methods found by Hitsch and Misra (2018).

Our paper contributes to several rapidly growing literatures, and we can not hope to do justice to each here. We give only those citations of particular relevance; more references can be found within these works. First, there has been much recent study of the statistical properties of machine learning tools as an end in itself. Many studies have focused on the lasso and its variants (Belloni et al., 2014; Bickel et al., 2009; Farrell, 2015) and tree/forest based methods (Wager and Athey, 2018). Relatively less work has been done for deep neural networks. An important exception is the recent work of Schmidt-Hieber (2019), who showed that a particular deep ReLU network with uniformly bounded weights attains the optimal rate in expected risk for squared loss. Further, Schmidt-Hieber (2019) formally shows that deep neural networks can strictly improve on classical methods: if the unknown target function is itself a composition of simpler functions, then the composition-based deep net estimator is provably superior to estimators that do not use compositions. This is a possible first step in theoretically understanding why deep learning is so successful empirically. Our work differs substantially from Schmidt-Hieber (2019). First, our goal is not to demonstrate adaptation, and we do not study this property of deep nets, but focus on the common nonparametric case. Second, our results and assumptions are quite different in that: (i) we prove nonasymptotic high probability bounds instead of bounds on the expected risk, (ii) we cover general, nonlinear regression problems, (iii) in linear models we allow for non-Gaussian, heteroskedastic errors, relying only on boundedness, and (iv) we allow for unbounded weight parameters, which is crucial for feasible implementation and for approximation results. Finally, our method of proof is entirely different from Schmidt-Hieber (2019), and it is this proof which enables us to deliver (i)–(iv). Specializing our results to the linear model treated by Schmidt-Hieber (2019), and looking at smooth functions, our high probability bounds imply expect risk bounds similar to those obtained in that paper, but under somewhat different regularity conditions: Schmidt-Hieber (2019) requires bounded weights and errors that are Guassian, independent of the covariates, and have known homoskedasticity. These differences between our work and Schmidt-Hieber (2019) are elaborated on further below, after stating our main results. Bach (2017) and Bauer and Kohler (2019) also make important contributions on adaptation properties of deep nets on functions with certain low dimensional structure. Yarotsky (2017, 2018) and Bartlett et al. (2017) are important building blocks for our results.

A second strand of literature focuses on inference following of machine learning. Initial theoretical results were concerned with obtaining valid inference on a coefficient in a high-dimensional regression, following model selection or regularization, with particular focus on the lasso (Belloni et al., 2014; Javanmard and Montanari, 2014; van de Geer et al., 2014). Intuitively, this is a semiparametric problem, where the coefficient of interest is estimable at the parametric rate and the remaining coefficients are collectively a nonparametric nuisance parameter estimated using machine learning methods. Building on this intuition, many have studied the semiparametric stage directly, such as obtaining novel, weaker conditions easing the application of machine learning methods (Belloni et al., 2014; Chernozhukov et al., 2018; Farrell, 2015, and references therein). We builds on this work, employing conditions therein, and in particular, verifying them for deep ReLU nets.

The next section introduces deep ReLU networks and states our main theoretical results: nonasymptotic bounds for nonparametric regression-type losses. Semiparametric inference is discussed in Section 3. The empirical study is in Section 4. Section 5 concludes and proofs are given in the appendix. We will use the following norms: for a random vector $\boldsymbol{X} \in \mathbb{R}^d$, with generic realization $\boldsymbol{x}$ and sample realization $\boldsymbol{x}_i$, and a function $g(\boldsymbol{x})$, $\|g\|_\infty := \sup_{\boldsymbol{x}} |g(\boldsymbol{x})|$, $\|g\|_{L_2(\boldsymbol{X})} := \mathbb{E}[g(\boldsymbol{X})^2]^{1/2}$, and $\|g\|_n := \mathbb{E}_n[g(\boldsymbol{x}_i)^2]^{1/2}$, where $\mathbb{E}_n[\cdot]$ denotes the sample average.

2. Deep Neural Networks

In this section we will give our main theoretical results: high-probability nonasymptotic bounds for deep neural network estimation. The utility of these results for second-step semiparametric causal inference (the downstream task), for which the implied convergence rates are sufficiently rapid, is demonstrated in Section 3. We view our results as an initial step in establishing both the estimation and inference theory for modern deep learning, i.e. neural networks built using the multi-layer perceptron architecture (described below) and the nonsmooth ReLU activation function. This combination is crucial: it has demonstrated state of the art performance empirically and can be feasibly optimized. This is in contrast with sigmoid-based networks, either shallow (for which theory exists, but may not match empirical performance) or deep (which are not feasible to optimize), and with shallow ReLU networks, which may not approximate broad function classes.

As neural networks are perhaps less familiar to economists and other social scientists, we first briefly review the construction of deep ReLU nets. Our main focus will be on the fully connected feedfoward neural network, frequently referred to as a multi-layer perceptron, as this may be the most commonly implemented network architecture and we want our results to inform empirical practice. However, our results are more general, accommodating other architectures provided they are able to yield a universal approximation (in the appropriate function class), and so we review neural nets more generally and give concrete examples.

Our goal is to estimate an unknown function $f_*(\boldsymbol{x})$ that relates the covariates $\boldsymbol{X} \in \mathbb{R}^d$ to a scalar outcome

Y as the minimizer of the expectation of a per-observation loss function. Collecting these random variables into the vector $\boldsymbol{Z} = (Y, \boldsymbol{X}')' \in \mathbb{R}^{d+1}$, with $\boldsymbol{z} = (y, \boldsymbol{x}')'$ denoting a realization, we write

$$f_* = \arg\min_f \mathbb{E}\left[\ell\left(f, \boldsymbol{Z}\right)\right].$$

We allow for any loss function that is Lipschitz in f and obeys a curvature condition around f_*. Specifically, for constants c_1, c_2, and C_ℓ that are bounded and bounded away from zero, we assume that $\ell(f, \boldsymbol{z})$ obeys

$$\begin{aligned} &|\ell(f,\boldsymbol{z}) - \ell(g,\boldsymbol{z})| \leq C_\ell |f(\boldsymbol{x}) - g(\boldsymbol{x})|, \\ &c_1 \mathbb{E}\left[(f - f_*)^2\right] \leq \mathbb{E}[\ell(f,\boldsymbol{Z})] - \mathbb{E}[\ell(f_*,\boldsymbol{Z})] \leq c_2 \mathbb{E}\left[(f - f_*)^2\right]. \end{aligned} \tag{2.1}$$

Our results will be stated for a general loss obeying these two conditions.[1] We give a unified localization analysis of all such problems. This family of loss function covers many interesting problems. Two leading examples, used in our application to causal inference, are least squares and logistic regression, corresponding to the outcome and propensity score models respectively. For least squares, the target function and loss are

$$f_*(\boldsymbol{x}) := \mathbb{E}[Y|\boldsymbol{X} = \boldsymbol{x}] \qquad \text{and} \qquad \ell\left(f, \boldsymbol{z}\right) = \frac{1}{2}(y - f(\boldsymbol{x}))^2, \tag{2.2}$$

respectively, while for logistic regression these are

$$f_*(\boldsymbol{x}) := \log \frac{\mathbb{E}[Y|\boldsymbol{X} = \boldsymbol{x}]}{1 - \mathbb{E}[Y|\boldsymbol{X} = \boldsymbol{x}]} \qquad \text{and} \qquad \ell\left(f, \boldsymbol{z}\right) = -yf(\boldsymbol{x}) + \log\left(1 + e^{f(\boldsymbol{x})}\right). \tag{2.3}$$

Lemma 8 verifies, with explicit constants, that (2.1) holds for these two. Losses obeying (2.1) extend beyond these cases to other generalized linear models, such as count models, and can even cover multinomial logistic regression (multiclass classification), as shown in Lemma 9.

2.1. Neural Network Constructions

We now briefly describe deep ReLU neural networks, paying closer attention to the details germane to our theory. Goodfellow et al. (2016) gives a complete introduction and many references.

The crucial choice is the specific network architecture, or class. In general we will call this $\mathcal{F}_{\text{DNN}}$. From a theoretical point of view, different classes have different complexity and different approximating power. We give results for several concrete examples below. We will focus on *feedforward neural networks.* An example of a feedforward network is shown in Figure 1. The network consists of d input units, corresponding to the covariates $\boldsymbol{X} \in \mathbb{R}^d$, one output unit for the outcome Y. Between these are U hidden units, or computational nodes or neurons. These are connected by a directed acyclic graph specifying the architecture. The key graphical feature of a feedforward network is that hidden units are grouped in a sequence of L layers, the *depth* of the network, where a node is in layer $l = 1, 2, \ldots, L$, if it has a predecessor in layer $l - 1$ and no predecessor in any layer $l' \geq l$. The *width* of the network at a given layer, denoted H_l, is the number of units in that layer. The network is completed with the choice of an *activation function* $\sigma : \mathbb{R} \mapsto \mathbb{R}$ applied to the output of each node as described below. In this paper, we focus on the popular ReLU activation function $\sigma(x) = \max(x, 0)$, though our results can be extended (at notational cost) to cover piecewise linear activation functions (see also Remark 3).

An important and widely used subclass is the one that is *fully connected* between consecutive layers but has *no* other connections and each layer has number of hidden units that are of the same order of magnitude. This architecture is often referred to as a *Multi-Layer Perceptron* (MLP) and we denote this class as $\mathcal{F}_{\text{MLP}}$. See Figure 2, cf. Figure 1. We will assume that all the width of all layers share a common asymptotic order H, implying that for this class $U \asymp LH$.

We allow for generic feedforward networks, but we present special results for the MLP case, as it is widely used in empirical practice. As we will see below, the architecture, through its complexity, and equally importantly, approximation power, plays a crucial role in the final bound. In particular, we find only a suboptimal rate for the MLP case, but our upper bound is still sufficient for semiparametric inference.

[1] We thank an anonymous referee for suggesting this approach of exposition.

To build intuition on the computation, and compare to other nonparametric methods, let us focus on least squares for the moment, i.e. Equation (2.2), with a continuous outcome using a multilayer perceptron with constant width H. Each hidden unit u receives an input in the form of a linear combination $\tilde{\boldsymbol{x}}'\boldsymbol{w} + b$, and then returns $\sigma(\tilde{\boldsymbol{x}}'\boldsymbol{w} + b)$, where the vector $\tilde{\boldsymbol{x}}$ collects the output of all the units with a directed edge into u (i.e., from prior layers), $\boldsymbol{w}$ is a vector of weights, and b is a constant term. (The constant term is often referred to as the "bias" in the deep learning literature, but given the loaded meaning of this term in inference, we will largely avoid referring to b as a bias.) To be precise, let $\tilde{x}_{h,l}$ denote the scalar output of a node $u = (h, l)$, for $h = 1, \dots H_l$, $l = 1, \dots L$, and let $\tilde{\boldsymbol{x}}_l = (\tilde{x}_{1,l}, \dots, \tilde{x}_{H,l})'$ for layer $l \leq L$. The full network is defined through recursion: each node computes $\tilde{x}_{h,l} = \sigma(\tilde{\boldsymbol{x}}'_{l-1}\boldsymbol{w}_{h,l-1} + b_{h,l-1})$ and the final output is $\widehat{y} = \widehat{f}_{\text{MLP}}(\boldsymbol{x}) = \tilde{\boldsymbol{x}}'_L\boldsymbol{w}_L + b_L$. The MLP estimator can be also written as a composition as follows. Define $\boldsymbol{W}_l$ as the $H_{l+1} \times H_l$ matrix collecting $\{\boldsymbol{w}_{h,l}\}_{h=1}^{H_l}$, where $H_0 = d$, $\boldsymbol{b}_l$ as the H_l-vector collecting $\{b_{h,l}\}_{h=1}^{H_l}$, and $\boldsymbol{\sigma} : \mathbb{R}^{H_l} \mapsto \mathbb{R}^{H_l}$ as the function which applies $\sigma(\cdot)$ component-wise. Then

$$\widehat{f}_{\text{MLP}}(\boldsymbol{x}) = \boldsymbol{W}_L\boldsymbol{\sigma}\Bigg(\cdots \boldsymbol{\sigma}\bigg(\boldsymbol{W}_3\boldsymbol{\sigma}\Big(\boldsymbol{W}_2\boldsymbol{\sigma}\big(\boldsymbol{W}_1\boldsymbol{\sigma}(\boldsymbol{W}_0\boldsymbol{x} + \boldsymbol{b}_0) + \boldsymbol{b}_1\big) + \boldsymbol{b}_2\Big) + \boldsymbol{b}_3\bigg) + \cdots \Bigg) + \boldsymbol{b}_L$$

(This exact structure does not hold for the more general case of Section 2.3.) It is also useful to write the output of the final layer as $\tilde{\boldsymbol{x}}_L = \tilde{\boldsymbol{x}}_L(\boldsymbol{x})$, explicitly as a function of the original covariates, and thus the final output may be seen as a basis function approximation (albeit a complex and data-dependent one), written as $\widehat{f}_{\text{MLP}}(\boldsymbol{x}) = \tilde{\boldsymbol{x}}_L(\boldsymbol{x})'\boldsymbol{w}_L + b_L$, which is reminiscent of a traditional series (linear sieve) estimator. If all layers save the last were fixed, we could simply optimize using least squares directly: $(\boldsymbol{w}_L, b_L) \in \arg\min_{\boldsymbol{w},b} \|y_i - \tilde{\boldsymbol{x}}'_L\boldsymbol{w} - b\|_n^2$.

The crucial distinction is that the basis functions $\tilde{\boldsymbol{x}}_L(\cdot)$ are learned from the data. The "basis" is $\tilde{\boldsymbol{x}}_L = (\tilde{x}_{1,L}, \dots, \tilde{x}_{H,L})'$, where each $\tilde{x}_{h,L} = \sigma(\tilde{\boldsymbol{x}}'_{L-1}\boldsymbol{w}_{h,L-1} + b_{h,L-1})$. Therefore, "before" we can solve the least squares problem above, we would have to estimate $(\boldsymbol{w}'_{h,L-1}, b_{h,L-1}), h = 1, \dots, H$, anticipating the final estimation. These in turn depend on the prior layer, and so forth back to the original inputs $\boldsymbol{x}$. Optimization proceeds layer-by-layer using (variants of) stochastic gradient descent, with gradients of the parameters calculated by back-propagation (implementing the chain rule) induced by the network structure. Our results match standard optimization methods by *not* requiring the weight parameters to be uniformly bounded. The collection, over all nodes, of $\boldsymbol{w}$ and b, constitutes the parameters θ which are optimized in the final estimation. We denote W as the total number of parameters of the network. For the MLP, $W = (d+1)H + (L-1)(H^2 + H) + H + 1$.

To further clarify the use of deep nets, it is useful to make explicit analogies to more classical nonparametric techniques, leveraging the form $\widehat{f}_{\text{MLP}}(\boldsymbol{x}) = \tilde{\boldsymbol{x}}_L(\boldsymbol{x})'\boldsymbol{w}_L + b_L$. For a traditional series estimator (such as splines) the two choices for the practitioner are the basis (the spline shape and degree) and the number of terms (knots), commonly referred to as the smoothing and tuning parameters, respectively. In kernel regression, these would respectively be the shape of the kernel (and degree of local polynomial) and the bandwidth(s). For neural networks, the same phenomena are present: the architecture as a *whole* (the graph structure and activation function) are the smoothing parameters while the width and depth play the role of tuning parameters.

The architecture plays a crucial role in that it determines the approximation power of the network, and it is worth noting that because of the relative complexity of neural networks, such approximations, and comparisons across architectures, are not simple. It is comparatively obvious that quartic splines are more flexible than cubic splines (for the same number of knots) as is a higher degree local polynomial (for the same bandwidth). At a glance, it may not be clear what function class a given network architecture (width, depth, graph structure, and activation function) can approximate. As we will show below, the MLP architecture is not yet known to yield an optimal approximation (for a given width and depth) and therefore we are only able to prove a bound with slower than optimal rate. As a final note, computational considerations are important for deep nets in a way that is not true conventionally; see Remarks 1, 2, and 3.

Just as for classical nonparametrics, for a fixed architecture it is the tuning parameters that determine the rate of convergence (fixing smoothness of f_*). The recent wave of theoretical study of deep learning is still in its infancy. As such, there is no understanding of optimal architectures or tuning parameters. These choices can be difficult and only preliminary research has been done (see e.g., Daniely, 2017; Telgarsky, 2016, and references therein). However, it is interesting that in some cases, results can be obtained even with a fixed width H, provided the network is deep enough; see Corollary 2.

In sum, for a user-chosen architecture $\mathcal{F}_{\mathrm{DNN}}$, encompassing the choices $\sigma(\cdot)$, U, L, W, and the graph structure, the final estimate is computed using observed samples $\boldsymbol{z}_i = (y_i, \boldsymbol{x}_i')'$, $i = 1, 2, \ldots, n$, of $\boldsymbol{Z}$, by solving

$$\widehat{f}_{\mathrm{DNN}} \in \underset{\substack{f_\theta \in \mathcal{F}_{\mathrm{DNN}} \\ \|f_\theta\|_\infty \leq 2M}}{\arg\min} \sum_{i=1}^{n} \ell(f, \boldsymbol{z}_i) \, . \tag{2.4}$$

Recall that θ collects, over all nodes, the weights and constants $\boldsymbol{w}$ and b. When (2.4) is restricted to the MLP class we denote the resulting estimator $\widehat{f}_{\mathrm{MLP}}$. The choice of M may be arbitrarily large, and is part of the definition of the class $\mathcal{F}_{\mathrm{DNN}}$. This is neither a tuning parameter nor regularization in the usual sense: it is not assumed to vary with n, and beyond being finite and bounding $\|f_*\|_\infty$ (see Assumption 1), no properties of M are required. This is simply a formalization of the requirement that the optimizer is not allowed to diverge on the function level in the l_∞ sense– the weakest form of constraint. It is important to note that while typically regularization will alter the approximation power of the class, that is not the case with the choice of M as we will assume that the true function $f_*(\boldsymbol{x})$ is bounded, as is standard in nonparametric analysis. With some extra notational burden, one can make the dependence of the bound on M explicit, though we omit this for clarity as it is not related to statistical issues.

Remark 1 *In applications it is common to apply some form of regularization to the optimization of (2.4). However, in theory, the role of explicit regularization is unclear and may be unnecessary, as stochastic gradient descent presents good, if not better, solutions empirically (Zhang et al., 2016). Explicit regularization may improve empirical performance in low signal-to-noise ratio problems. There are many alternative regularization methods, including L_1 and L_2 (weight decay) penalties, drop out, and others, a detailed investigation of which is beyond the present scope.*

2.2. Nonasymptotic High-Probability Bounds for Multi-Layer Perceptrons

We now state our main theoretical results: nonasymptotic high-probability bounds for deep ReLU networks. We begin by discussing our assumptions. The sampling assumptions we require are collected in the following.

Assumption 1 *Assume that $\boldsymbol{z}_i = (y_i, \boldsymbol{x}_i')', 1 \leq i \leq n$ are i.i.d. copies of $\boldsymbol{Z} = (Y, \boldsymbol{X}) \in \mathcal{Y} \times [-1, 1]^d$, where X is continuously distributed. For an absolute constant $M > 0$, assume $\|f_*\|_\infty \leq M$ and $\mathcal{Y} \subset [-M, M]$.*

This assumption is fairly standard in nonparametrics. The only restriction worth mentioning is that the outcome is bounded. In many cases this holds by default (such as logistic regression, where $\mathcal{Y} = \{0, 1\}$) or count models (where $\mathcal{Y} = \{0, 1, \ldots, M\}$, with M limited by real-world constraints). For continuous outcomes, such as least squares regression, our restriction is not substantially more limiting than the usual assumption of a model such as $Y = f_*(\boldsymbol{X}) + \varepsilon$, where $\boldsymbol{X}$ is compact-supported, f_* is bounded, and the stochastic error ε possesses many moments. Indeed, in many applications such a structure is only coherent with bounded outcomes, such as the common practice of including lagged outcomes as predictors. Next, the assumption of continuously distributed covariates is quite standard. Discrete covariates taking on many values may be more realistically thought of as continuous, and it may be more accurate to allow these to slow the convergence rates. Our focus on $L_2(\boldsymbol{X})$ convergence allows for these essentially automatically. Finally, from a practical point of view, deep networks handle discrete covariates seamlessly and have demonstrated excellent empirical performance, which is in contrast to other more classical nonparametric techniques that may require manual adaptation.

Our main result treats the multi-layer perceptron architecture, with the ReLU activation function and unbounded weights, matching perhaps the most standard deep neural network. Such MLPs are now known to approximate smooth functions well (Yarotsky, 2017, 2018), leading to our next assumption: that the target function f_* lies in a Hölder ball with certain smoothness. Discussion of Hölder, Sobolev, and Besov spaces can be found in Gine and Nickl (2016).

ASSUMPTION 2 *Assume f_* lies in the Hölder ball $\mathcal{W}^{\beta,\infty}([-1,1]^d)$, with smoothness $\beta \in \mathbb{N}_+$,*

$$f_*(x) \in \mathcal{W}^{\beta,\infty}([-1,1]^d) := \left\{ f : \max_{\alpha, |\alpha| \le \beta} \operatorname*{ess\,sup}_{x \in [-1,1]^d} |D^\alpha f(x)| \le 1 \right\},$$

where $\alpha = (\alpha_1, \ldots, \alpha_d)$, $|\alpha| = \alpha_1 + \ldots + \alpha_d$ and $D^\alpha f$ is the weak derivative.

Under Assumptions 1 and 2 we obtain the following high probability bounds, covering a host of models, which, to the best of our knowledge, is new to the literature. In some sense, this is our main result for deep learning, as it deals with the most common architecture.

THEOREM 1 (Multi-Layer Perceptron) *Suppose Assumptions 1 and 2 hold. Let $\widehat{f}_{\mathrm{MLP}}$ be the deep MLP-ReLU network estimator defined by (2.4), restricted to $\mathcal{F}_{\mathrm{MLP}}$, for a loss function obeying (2.1), with width $H \asymp n^{\frac{d}{2(\beta+d)}} \log^2 n$ and depth $L \asymp \log n$. Then with probability at least $1 - \exp(-n^{\frac{d}{\beta+d}} \log^8 n)$, for n large enough,*

(a) $\|\widehat{f}_{\mathrm{MLP}} - f_*\|^2_{L_2(\boldsymbol{X})} \le C \cdot \left\{ n^{-\frac{\beta}{\beta+d}} \log^8 n + \dfrac{\log\log n}{n} \right\}$ *and*

(b) $\mathbb{E}_n\left[(\widehat{f}_{\mathrm{MLP}} - f_*)^2 \right] \le C \cdot \left\{ n^{-\frac{\beta}{\beta+d}} \log^8 n + \dfrac{\log\log n}{n} \right\}$,

for a constant $C > 0$ independent of n, which may depend on d, M, and other fixed constants.

Several aspects of these nonasymptotic bound warrant discussion. We build on the recent results of Bartlett et al. (2017), who find nearly-tight bounds on the pseudo-dimension of deep nets. One contribution of our proof is to use a *scale sensitive* localization theory (Bartlett et al., 2005; Koltchinskii, 2006, 2011; Koltchinskii and Panchenko, 2000; Liang et al., 2015) with such *scale insensitive* measures for deep neural networks for general smooth loss functions. This has two tangible benefits. First, we do not restrict the class of network architectures to have bounded weights for each unit (scale insensitive), in accordance to standard practice (Zhang et al., 2016) wherein optimization is not constrained, and in contrast to the classic sieve analysis with scale sensitive measure such as metric entropy. This allows for a richer set of approximating possibilities, in particular allowing more flexibility in seeking architectures with specific properties, as we explore in the next subsection. For the special case of least squares regression, Koltchinskii (2011) uses a similar approach, and a similar result to our Theorem 1(a) can be derived for this case using his Theorem 5.2 and Example 3 (p. 85f). This is perhaps the nearest antecedent to our result. To avoid repetition, other important results are discussed following Theorem 2 below.

Second, we are able to attain a faster rate on the second term of the bound, order n^{-1} in the sample size, instead of the $n^{-1/2}$ that would result from a direct application of uniform deviation bounds. This upper bound informs the trade offs between H and L, and the approximation power, and may point toward optimal architectures for statistical inference. Even with these choices of H and L, the bound of Theorem 1 is not optimal (for fixed β, in the sense of Stone (1982)). We rely on the explicit approximating constructions of Yarotsky (2017), and it is possible that in the future improved approximation properties of MLPs will be found, allowing for a sharpening of the results of Theorem 1 immediately, i.e. without change to our theoretical argument. At present, it is not clear if this rate can be improved, but it is sufficiently fast for valid inference.

Finally, we note that as is standard in nonparametrics, this result relies on choosing H appropriately given the smoothness β of Assumption 2. Of course, the true smoothness is unknown and thus in practice the "β" appearing in H, and consequently in the convergence rates, need not match that of Assumption 2. In general, the rate will depend on the smaller of the two. Most commonly it is assumed that the user-chosen β is fixed and that the truth is smoother; witness the ubiquity of cubic splines and local linear regression. Rather than spell out these consequences directly, we will tacitly assume the true smoothness is not less than the β appearing in H (here and below). Smoothness adaptive approaches, as in classical nonparametrics, may also be possible with deep nets, but are beyond the scope of this study.

2.3. Other Network Architectures

Theorem 1 covers only one specific architecture, albeit the most important one for current practice. However, given that this field is rapidly evolving, it is important to consider other possible architectures which may be beneficial in some cases. To this end, we will state a more generic result and then two specific examples: one to obtain a faster rate of convergence and one for fixed-width networks. All of these results are, at present, more of theoretical interest than practical value, as they are either agnostic about the network (thus infeasible) or rely on more limiting assumptions.

In order to be agnostic about the specific architecture of the network we need to be flexible in the approximation power of the class. To this end, we will replace Assumption 2 with the following generic assumption, rather more of a definition, regarding the approximation power of the network.

Assumption 3 *Let f_* lie in a class $\mathcal{F}$. For the feedforward network class $\mathcal{F}_{\text{DNN}}$, used in (2.4), let the approximation error ϵ_{DNN} be*

$$\epsilon_{\text{DNN}} := \sup_{f_* \in \mathcal{F}} \inf_{\substack{f \in \mathcal{F}_{\text{DNN}} \\ \|f\|_\infty \leq 2M}} \|f - f_*\|_\infty \ .$$

It may be possible to require only an approximation in the $L_2(\boldsymbol{X})$ norm, but this assumption matches the current approximation theory literature and is more comparable with other work in nonparametrics, and thus we maintain the uniform definition. We then obtain the following result.

Theorem 2 (General Feedforward Architecture) *Suppose Assumptions 1 and 3 hold. Let $\widehat{f}_{\text{DNN}}$ be the deep ReLU network estimator defined by (2.4), for a loss function obeying (2.1). Then with probability at least $1 - e^{-\gamma}$, for n large enough,*

(a) $\|\widehat{f}_{\text{DNN}} - f_*\|^2_{L_2(\boldsymbol{X})} \leq C \left(\frac{WL \log W}{n} \log n + \frac{\log \log n + \gamma}{n} + \epsilon^2_{\text{DNN}} \right)$ *and*

(b) $\mathbb{E}_n \left[(\widehat{f}_{\text{DNN}} - f_*)^2 \right] \leq C \left(\frac{WL \log W}{n} \log n + \frac{\log \log n + \gamma}{n} + \epsilon^2_{\text{DNN}} \right)$,

for a constant $C > 0$ independent of n, which may depend on d, M, and other fixed constants.

This result is more general than Theorem 1, covering the general deep ReLU network problem defined in (2.4), general feedforward architectures, and the general class of losses defined by (2.1). The same comments as were made following Theorem 1 apply here as well: the same localization argument is used with the same benefits. We explicitly use this in the next two corollaries, where we exploit the allowed flexibility in controlling ϵ_{DNN} by stating results for particular architectures. The bound here is not directly applicable without specifying the network structure, which will determine both the variance portion (through W, L, and U) and the approximation error. With these set, the bound becomes operational upon choosing γ, which can be optimized as desired.

Perhaps the most directly related existing result, in addition to the aforementioned result of Koltchinskii (2011), is Theorem 2 of Schmidt-Hieber (2019), which also uses generic approximation error. That result is not a high-probability bound, only a rate on the expected risk, only covers squared loss, and requires Gaussian noise that is independent of the covariates and has known, homoskedastic variance, and, importantly, requires uniformly bounded weights in the network. The assumption of bounded weights may be difficult to impose computational and can limit the approximation power of the network. To see this last point consider a simple example: suppose that $d = 1$ and $f_*(x) = \sigma(\zeta x + 1)/2 - \sigma(\zeta x - 1)/2$. This f_*, for any ζ, is bounded and can be realized by a ReLU network without norm constraints using only two hidden units, and is thus estimable at $1/n$. However, for $\zeta > 1$ a network with weights bounded by one (as in Schmidt-Hieber (2019)) must have width 2ζ, so ζ must be known, and yields expected risk of order ζ/n.

Turning to special cases, we first show that the optimal rate of Stone (1982) can be attained, up to log factors. However, this relies on a rather artificial network structure, designated to approximate functions in a Sobolev space well, but without concern for practical implementation. Thus, while the following rate improves upon Theorem 1, we view this result as mainly of theoretical interest: establishing that (certain) deep ReLU networks are able to attain the optimal rate.

COROLLARY 1 (Optimal Rate) *Suppose Assumptions 1 and 2 hold. Let $\widehat{f}_{\text{OPT}}$ solve (2.4) using the (deep and wide) network of Yarotsky (2017, Theorem 1), with $W \asymp U \asymp n^{\frac{d}{2\beta+d}} \log n$ and depth $L \asymp \log n$, the following hold with probability at least $1 - e^{-\gamma}$, for n large enough,*

(a) $\|\widehat{f}_{\text{OPT}} - f_*\|^2_{L_2(\boldsymbol{X})} \leq C \cdot \left\{ n^{-\frac{2\beta}{2\beta+d}} \log^4 n + \frac{\log\log n + \gamma}{n} \right\}$ *and*

(b) $\mathbb{E}_n \left[(\widehat{f}_{\text{OPT}} - f_*)^2 \right] \leq C \cdot \left\{ n^{-\frac{2\beta}{2\beta+d}} \log^4 n + \frac{\log\log n + \gamma}{n} \right\}$,

for a constant $C > 0$ independent of n, which may depend on d, M, and other fixed constants.

The same rate, up to log factors, albeit concerning only the expected risk and subject to the other limitations above, can be obtained from Theorems 2 and 5 of Schmidt-Hieber (2019). However, the main goal of Schmidt-Hieber (2019) is not the standard nonparametric problem considered here, but rather in studying dimension adaptivity. Specifically, the main result therein, Theorem 1, shows that if f_* is itself a composition of functions which are individually estimable faster than $n^{-\frac{2\beta}{2\beta+d}}$, then a *sparsely* connected deep ReLU network adapts to this structure and attains the faster rate, an oracle type result. We do not explicitly study sparse networks. Further, it is shown that estimators which are not based on a composition structure do not possess the same adaptation property. For more on the results and limitations of Schmidt-Hieber (2019), see the published discussions (Ghorbani et al., 2019; Kutyniok, 2019; Shamir, 2019). Other work in this direction is Bach (2017) and Bauer and Kohler (2019). Polson and Ročková (2018) also obtain bounds for deep nets, building on these works, but applied in a Bayesian context.

Next, we turn to *very* deep networks that are very narrow, which have attracted substantial recent interest. Theorem 1 and Corollary 1 dealt with networks where the depth and the width grow with sample size. This matches the most common empirical practice, and is what we use in Sections 4. However, it is possible to allow for networks of *fixed* width, provided the depth is sufficiently large. The next result is perhaps the largest departure from the classical study of neural networks: earlier work considered networks with diverging width but fixed depth (often a single layer), while the reverse is true here. The activation function is of course qualitatively different as well, being piecewise linear instead of smooth. Using recent results (Hanin, 2017; Mhaskar and Poggio, 2016; Yarotsky, 2018) we can establish the following rate for very deep, fixed-width MLPs.

COROLLARY 2 (Fixed Width Networks) *Let the conditions of Theorem 1 hold, with $\beta \geq 1$ in Assumption 2. Let $\widehat{f}_{\text{FW}}$ solve (2.4) for an MLP with fixed width $H = 2d + 10$ and depth $L \asymp n^{\frac{d}{2(2+d)}}$. Then with probability at least $1 - e^{-\gamma}$, for n large enough,*

(a) $\|\widehat{f}_{\text{FW}} - f_*\|^2_{L_2(\boldsymbol{X})} \leq C \cdot \left\{ n^{-\frac{2}{2+d}} \log^2 n + \frac{\log\log n + \gamma}{n} \right\}$ *and*

(b) $\mathbb{E}_n \left[(\widehat{f}_{\text{FW}} - f_*)^2 \right] \leq C \cdot \left\{ n^{-\frac{2}{2+d}} \log^2 n + \frac{\log\log n + \gamma}{n} \right\}$,

for a constant $C > 0$ independent of n, which may depend on d, M, and other fixed constants.

This result is again mainly of theoretical interest. The class is only able to approximate well functions with $\beta = 1$ (cf. the choice of L) which limits the potential applications of the result because, in practice, d will be large enough to render this rate, unlike those above, too slow for use in later inference procedures. In particular, if $d \geq 3$, the sufficient conditions of Theorem 3 fail.

Finally, as mentioned following Theorem 1, our theory here will immediately yield a faster rate upon discovery of improved approximation power of this class of networks. In other words, for example, if a proof became available that fixed-width, very deep networks can approximate β-smooth functions (as in Assumption 2), then Corollary 2 will trivially be improvable to match the rate of Theorem 1. Similarly, if the MLP architecture can be shown to share the approximation power with that of Corollary 1, then Theorem 1 will itself deliver the optimal rate. Our proofs will not require adjustment.

REMARK 2 *Although there has been a great deal of work in easing implementation (optimization and tuning) of deep nets, it still may be a challenge in some settings, particularly when using non-standard architectures. See also Remark 1. Given the renewed interest in deep networks, this is an area of study already (Hartford* *et al., 2017; Polson and Ročková, 2018) and we expect this to continue and that implementations will rapidly evolve. This is perhaps another reason that Theorem 1 is, at the present time, the most practically useful, but that (as just discussed) Theorem 2 will be increasingly useful in the future.*

Remark 3 *Our results can be extended easily to include piecewise linear activation functions beyond ReLU, using the complexity result obtained in Bartlett et al. (2017). In principle, similar rates of convergence could be attained for other activation functions, given results on their approximation error. However, it is not clear what practical value would be offered due to computational issues (in which the activation choice plays a crucial role). Indeed, the recent switch to ReLU stems not from their greater approximation power, but from the fact that optimizing a deep net with sigmoid-type activation is unstable or impossible in practice. Thus, while it is certainly possible that we could complement the single-layer results with rates for sigmoid-based deep networks, these results would have no consequences for real-world practice.*

From a purely practical point of view, several variations of the ReLU activation function have been proposed recently (including the so-called Leaky ReLU, Randomized ReLU, (Scaled) Exponential Linear Units, and so forth) and have been found in some experiments to improve optimization properties. It is not clear what theoretical properties these activation functions have or if the computational benefits persist more generically, though this area is rapidly evolving. We conjecture that our results could be extended to include these activation functions.

3. Inference After Deep Learning

We will use the results above, in particular Theorem 1, coupled with results in the semiparametric literature, to deliver valid asymptotic inference following deep learning. The novelty of our results is not in this semiparametric stage per se, but rather in delivering valid inference after deep learning, and therefore our discussion will be brief. Our results for deep learning can be applied much more generally, see the longer version of this paper (Farrell et al., 2019a) and other recent literature (Belloni et al., 2017; Chernozhukov et al., 2018). At present, we focus on average causal parameters here, as they are popular both in applications and in theoretical work, so our results can be put to immediate use as well as easily compared to prior literature.

To briefly describe the set up: we observe a sample of n units, each exposed to a binary treatment, and for each unit we observe a vector of pre-treatment covariates, $\boldsymbol{X} \in \mathbb{R}^d$, treatment status $T \in \{0,1\}$, and a scalar post-treatment outcome Y. The observed outcome obeys $Y = TY(1) + (1-T)Y(0)$, where $Y(t)$ is the unobserved potential outcome under treatment status $t \in \{0,1\}$. The prototypical parameter of interest is the average treatment effect, $\tau := \mathbb{E}[Y(1) - Y(0)]$, also referred to as "lift" in digital context. We also study $\pi(s) := \mathbb{E}[s(\boldsymbol{X})Y(1) + (1-s(\boldsymbol{X}))\,Y(0)]$, the average realized outcome from a *counterfactual* treatment policy, $s(\boldsymbol{x}) : \operatorname{supp}\{\boldsymbol{X}\} \mapsto \{0,1\}$, that assigns a given set of characteristics (e.g. a consumer profile) to treatment status. Note well that this is *not* necessarily the observed treatment: $s(\boldsymbol{x}_i) \neq t_i$. Intuitively, τ is the expected gain from treating the "next" person, relative to if they had not been exposed, i.e., the expected *change* in the outcome. On the other hand, $\pi(s)$, which is the expected utility, welfare, or profit, is concerned with the *total* outcome that would be observed for the next person if the treatment rule were $s(\boldsymbol{x})$. The parameter depends on a counterfactual/hypothetical treatment targeting strategy, which is often itself the object of evaluation.

We make the following standard assumption of unconfoundedness and overlap, which delivers identification of the average treatment effect and, at no additional cost, counterfactual welfare.

Assumption 4 *Let $p(\boldsymbol{x}) = \mathbb{P}[T = 1|\boldsymbol{X} = \boldsymbol{x}]$ denote the propensity score and $\mu_t(\boldsymbol{x}) = E[Y(t)|\boldsymbol{X} = \boldsymbol{x}]$, $t \in \{0,1\}$ denote the two outcome regression functions. For $t \in \{0,1\}$ and almost surely $\boldsymbol{X}$, $\mathbb{E}[Y(t)|T, \boldsymbol{X} = \boldsymbol{x}] = \mathbb{E}[Y(t)|\boldsymbol{X} = \boldsymbol{x}]$ and $\bar{p} \leq p(\boldsymbol{x}) \leq 1 - \bar{p}$ for some $\bar{p} > 0$.*

Our approach to inference follows the current literature and uses sample averages of the (uncentered) influence functions. This approach yields valid inference under weaker conditions on the first step estimates (Chernozhukov et al., 2018; Farrell, 2015). Hahn (1998) shows that the influence function for a single average potential outcome is given by, $\psi_t(\boldsymbol{z}) - \mathbb{E}[Y(t)]$, for $t \in \{0,1\}$ and $\boldsymbol{z} = (y, t, \boldsymbol{x}')'$, where $\psi_t(\boldsymbol{z}) = \mathbb{1}\{T = t\}(y - \mu_t(\boldsymbol{x}))\mathbb{P}[T = t \mid \boldsymbol{X} = \boldsymbol{x}]^{-1} + \mu_t(\boldsymbol{x})$. We estimate the unknown functions with deep learning to form

$$\widehat{\psi}_t(\boldsymbol{z}_i) = \frac{\mathbb{1}\{t_i = t\}(y_i - \widehat{\mu}_t(\boldsymbol{x}_i))}{\widehat{\mathbb{P}}[T = t \mid \boldsymbol{X} = \boldsymbol{x}_i]} + \widehat{\mu}_t(\boldsymbol{x}_i), \tag{3.1}$$

where $\widehat{\mathbb{P}}[T = t \mid \boldsymbol{X} = \boldsymbol{x}_i] = \widehat{p}(\boldsymbol{x}_i)$ for $t = 1$ and $1 - \widehat{p}(\boldsymbol{x}_i)$ for $t = 0$. The final estimators of τ and $\pi(s)$ are obtained by taking appropriate linear combinations:

$$\widehat{\tau} = \mathbb{E}_n\left[\widehat{\psi}_1(\boldsymbol{z}_i) - \widehat{\psi}_0(\boldsymbol{z}_i)\right] \qquad \text{and} \qquad \widehat{\pi}(s) = \mathbb{E}_n\left[s(\boldsymbol{x}_i)\widehat{\psi}_1(\boldsymbol{z}_i) + (1 - s(\boldsymbol{x}_i))\widehat{\psi}_0(\boldsymbol{z}_i)\right]. \tag{3.2}$$

To add a per-unit cost of treatment/targeting c and a margin m, simply replace ψ_1 with $m\psi_1 - c$ and ψ_0 with $m\psi_0$. It may also be useful to compare a candidate targeting strategy, say $s'(\boldsymbol{x})$, to baseline or status quo policy, $s_0(\boldsymbol{x})$, by studying $\pi(s') - \pi(s_0) = \mathbb{E}\big[(s'(\boldsymbol{X}) - s_0(\boldsymbol{X}))Y(1) + (s_0(\boldsymbol{X}) - s'(\boldsymbol{X}))\,Y(0)\big] = \mathbb{E}[(s'(\boldsymbol{X}) - s_0(\boldsymbol{X}))\tau(\boldsymbol{X})]$, where $\tau(\boldsymbol{x}) = \mathbb{E}[Y(1) - Y(0) \mid \boldsymbol{X} = \boldsymbol{x}]$ is the conditional average treatment effect. The latter form makes clear that only those differently treated, of course, impact the evaluation of s' compared to s_0. The strategy s' will be superior if, on average, it targets those with a higher individual treatment effect.

We then obtain inference using the following result. Let β_p and β_μ be the smoothness parameters of Assumption 2 for the propensity score and outcome models, respectively.

Theorem 3 *Suppose that $\{\boldsymbol{z}_i = (y_i, t_i, \boldsymbol{x}_i')'\}_{i=1}^n$ are i.i.d. obeying Assumption 4 and the conditions Theorem 1 hold with $\beta_p \wedge \beta_\mu > d$. Further assume that, for $t \in \{0,1\}$, $\mathbb{E}[(s(\boldsymbol{X})\psi_t(\boldsymbol{Z}))^2|\boldsymbol{X}]$ is bounded away from zero and for some $\delta > 0$, $\mathbb{E}[(s(\boldsymbol{X})\psi_t(\boldsymbol{Z}))^{4+\delta}|\boldsymbol{X}]$ is bounded. Then the deep MLP-ReLU network estimators defined above obey the following, for $t \in \{0,1\}$, (a) $\mathbb{E}_n[(\widehat{p}(\boldsymbol{x}_i) - p(\boldsymbol{x}_i))^2] = o_P(1)$ and $\mathbb{E}_n\left[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))^2\right] = o_P(1)$, (b) $\mathbb{E}_n[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))^2]^{1/2}\mathbb{E}_n[(\widehat{p}(\boldsymbol{x}_i) - p(\boldsymbol{x}_i))^2]^{1/2} = o_P(n^{-1/2})$, and (c) $\mathbb{E}_n[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))(1 - \mathbb{1}\{t_i = t\}/\mathbb{P}[T = t|\boldsymbol{X} = \boldsymbol{x}_i])] = o_P(n^{-1/2})$, and therefore, if $\widehat{p}(\boldsymbol{x}_i)$ is bounded inside $(0,1)$, for a given $s(\boldsymbol{x})$ and $t \in \{0,1\}$, we have*

$$\sqrt{n}\mathbb{E}_n\left[s(\boldsymbol{x}_i)\widehat{\psi}_t(\boldsymbol{z}_i) - s(\boldsymbol{x}_i)\psi_t(\boldsymbol{z}_i)\right] = o_P(1) \qquad \text{and} \qquad \frac{\mathbb{E}_n[(s(\boldsymbol{x}_i)\widehat{\psi}_t(\boldsymbol{z}_i))^2]}{\mathbb{E}_n[(s(\boldsymbol{x}_i)\psi_t(\boldsymbol{z}_i))^2]} = o_P(1).$$

It is immediate from Theorem 3 that the estimators of (3.2), and other similar estimators, are asymptotically Normal with estimable variance. Looking at $\widehat{\pi}(s)$ to fix ideas,

$$\begin{aligned} &\sqrt{n}\widehat{\Sigma}^{-1/2}\left(\widehat{\pi}(s) - \pi(s)\right) \xrightarrow{d} \mathcal{N}(0,1), \\ &\quad\text{with} \quad \widehat{\Sigma} = \mathbb{E}_n\left[\left(s(\boldsymbol{x}_i)\widehat{\psi}_1(\boldsymbol{z}_i) + (1 - s(\boldsymbol{x}_i))\widehat{\psi}_0(\boldsymbol{z}_i)\right)^2\right] - \widehat{\pi}(s)^2. \end{aligned}$$

Further, Theorem 3 can be generalized immediately to yield uniformly valid inference (Belloni et al., 2014; Farrell, 2015). Finally, it is worth specializing this result to randomized experiments due to their popularity in practice, particularly important in the Internet age. In this case, the propensity score is estimated with the sample frequency, $\widehat{p}(\boldsymbol{x}_i) \equiv \widehat{p} = \mathbb{E}_n[t_i]$, and conditions (a) and (b) of Theorem 3 collapse, leaving only condition (c). The following is a trivial corollary of Theorem 3.

Corollary 3 *Let the conditions of Theorem 3 hold but instead of Assumption 4, assume T is independent of $Y(0)$, $Y(1)$, and X, and is distributed Bernoulli with parameter p^* bounded inside $(0,1)$. Then, if $\beta_\mu > d$, the deep MLP-ReLU networks obey* **(a′)** $\mathbb{E}_n\left[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))^2\right] = o_P(1)$ *and* **(c′)** $\mathbb{E}_n[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))(1 - \mathbb{1}\{t_i = t\}/p^*)] = o_P(n^{-1/2})$, *and the results of Theorem 3 hold.*

Theorem 3 shows, for a specific context, how deep learning delivers valid asymptotic inference for our parameters of interest. Theorem 1 (a generic result using Theorem 2 could be stated) proves that the nonparametric estimates converge sufficiently fast, as formalized by conditions (a), (b), and (c), enabling feasible efficient semiparametric inference. Proofs and further discussion of similar results can be found in

Chernozhukov et al. (2018); Farrell (2015). Here it is worth mentioning that the condition (c), which arises from a "leave-in" type remainder, can be weakened using sample splitting. Instead, we employ our localization analysis, as was used to obtain the results of Section 2, to verify (c) directly (see Lemma 10); this appears to be a novel application of localization, and this approach may be useful in future applications of second-step inference using machine learning methods where the theoretical gain of weaker requirements may not be worth the price paid in constants in finite samples.

Finally, we close this discussion by noting that our focus with Theorem 3 is showcasing the practical utility of deep learning in inference, and not in attaining minimal conditions. The requirement that $\beta_p \wedge \beta_\mu > d$, or $\beta_p \wedge \beta_\mu > d/2$ in Corollary 1, is not minimal. Minimal conditions for semiparametric inference have been studied by many, dating at least to Bickel and Ritov (1988); see Robins et al. (2009) for recent results and references. For causal inference, Chen et al. (2008) and Athey et al. (2018) obtain efficiency under weaker conditions than ours on $p(\boldsymbol{x})$ (the former under minimal smoothness on $\mu_t(\boldsymbol{x})$ and the latter under a sparsity in a high-dimensional linear model). Further, cross-fitting paired with local robustness may yield weaker smoothness conditions by providing underfitting" robustness, i.e., weakening bias-related assumptions (Chernozhukov et al., 2018), but the cost may be too high here. Weaker variance-related assumptions, or "overfitting" robustness (Cattaneo et al., 2018), may also be possible following deep learning, but are less automatic at present. Other methods for causal inference under relaxed assumptions may be useful here, such as extensions to doubly robust estimation (Tan, 2018) and inverse weighting (Ma and Wang, 2018).

4. Empirical Application

To illustrate our results we study a data from a large US retailer of consumer products. The firm sells directly to the customer (as opposed to via retailers) using a variety of channels such as the web and mail. Targeted marketing instruments, such as catalogs, aim to induce demand and often contain advertising and informational content about the firms offerings. It is important to carefully select which customers should be sent this material, i.e., be targeted for treatment, since the costs of its creation and dissemination accumulate rapidly. For a typical retailer the costs of one catalog may be close to a dollar. With millions of catalogs being sent, ascertaining the causal effects of such targeted mailing, and then using these effects to evaluate potential targeting strategies, is crucial for policy making. For a full discussion, see Hitsch and Misra (2018) (we use their 2015 sample).

The data consists of 292,657 consumers chosen at random from the retailer's database. Of these, 2/3 were randomly chosen to receive a catalog (the treatment). We observe treatment status, roughly one hundred fifty covariates, including demographics, past purchase behaviors, interactions with the firm, and other relevant information, and total consumer spending, the outcome of interest, aggregated from all available purchase channels including phone, mail, and the web, in a three-month window. Average spending is \$7.31, but for the roughly six percent who made a purchase, the average spend is \$117.73.

We implement Equations (3.1) and (3.2) for eight different deep nets. All computation was done using TensorFlow™. The details of the eight deep net architectures are presented in Table I. A key measure of fit reported in the final column of the table is the portion of $\widehat{\tau}(\boldsymbol{x}_i)$ that were negative. As argued by Hitsch and Misra (2018), it is implausible, for nearly all individuals, under standard marketing or economic theory that receipt of a catalog causes lower purchasing. Here deep nets perform as well as, and sometimes better than, the best methods found by Hitsch and Misra (2018). Figure 3 shows the distribution of $\widehat{\tau}(\boldsymbol{x}_i)$ across customers for each of the eight architectures. While there are differences in the shapes, the mean and variance estimates are nonetheless similar. We also conducted a placebo test: using only the untreated customers in the data and randomly assigning half to treated status we then re-ran the eight architectures.[2] Figure 4 plots $\widehat{\tau}(\boldsymbol{x}_i)$ for these, and we see that the "true" zero average effect is recovered and with the expected distribution.

Table II shows the estimates of the average treatment effect and the counterfactual profits from three different targeting strategies, along with their respective 95% confidence intervals. The strategies are (i) *never* treat, $s(\boldsymbol{x}) \equiv 0$; (ii) a *blanket* treatment, $s(\boldsymbol{x}) \equiv 1$; (iii) a *loyalty* policy, $s(\boldsymbol{x}) = 1$ only for those who had purchased in the prior calendar year. In all cases we add a profit margin m and a mailing cost c to $\pi(s)$ (our NDA with the firm forbids revealing m and c). It is clear that profits from the three policies are ordered

[2] We thank Guido Imbens suggesting this analysis.

as $\pi(\text{never}) < \pi(\text{blanket}) < \pi(\text{loyalty})$. In all results, there is broad agreement among the eight architectures. This may be due to the fact that the data is experimental, so that the propensity score is constant. We have explored this using simulations, which are reported in the Supplemental Material (Farrell et al., 2019b).

5. Conclusion

The utility of deep learning in social science applications is still a subject of interest and debate. While there is an acknowledgment of its predictive power, there has been limited adoption of deep learning in social sciences such as economics. Some part of the reluctance to adopting these methods stems from the lack of theory facilitating use and interpretation. We have shown, both theoretically as well as empirically, that these methods can offer excellent performance.

In this paper, we have given a formal proof that inference can be valid after using deep learning methods for first-step estimation. Our results thus contribute directly to the recent explosion in both theoretical and applied research using machine learning methods in economics, and to the recent adoption of deep learning in empirical settings. We obtained novel bounds for deep neural networks, speaking directly to the modern (and empirically successful) practice of using fully-connected feedfoward networks. Our results allow for different network architectures, including fixed width, very deep networks. Our results cover general nonparametric regression-type loss functions, covering most nonparametric practice. We used our bounds to deliver fast convergence rates allowing for second-stage inference on a finite-dimensional parameter of interest.

There are practical implications of the theory presented in this paper. We focused on semiparametric causal effects as a concrete illustration, but deep learning is a potentially valuable tool in many diverse economic settings. Our results allow researchers to embed deep learning into standard econometric models such as linear regressions, generalized linear models, and other forms of limited dependent variables models (e.g. censored regression). Our theory can also be used as a starting point for constructing deep learning implementations of two-step estimators in the context of selection models, dynamic discrete choice, and the estimation of games.

To be clear, we see our paper as an early step in the exploration of deep learning as a tool for economic applications. There are a number of opportunities, questions, and challenges that remain. For some estimands, it may be crucial to estimate the density as well, and this problem can be challenging in high dimensions. Deep nets, in the formulation of GANs, are a promising tool for distribution estimation (Athey et al., 2019; Liang, 2018). There are also interesting questions of network architectures representing, and adapting to, the underlying function, and if these can be learned from the data (Bach, 2017; Dou and Liang, 2020). Lastly, furter computational and optimization guidance is needed. Research into these applications and structures is underway.

6. References

Anthony, M. and P. L. Bartlett (1999): *Neural Network Learning: Theoretical Foundations*, Campbridge University Press.

Athey, S., G. W. Imbens, J. Metzger, and E. M. Munro (2019): "Using Wasserstein Generative Adversarial Networks for the Design of Monte Carlo Simulations," Tech. rep., National Bureau of Economic Research.

Athey, S., G. W. Imbens, and S. Wager (2018): "Approximate residual balancing: debiased inference of average treatment effects in high dimensions," *Journal of the Royal Statistical Society, Series B*, 80, 597–623.

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

Bartlett, P. L., O. Bousquet, and S. Mendelson (2005): "Local rademacher complexities," *The Annals of Statistics*, 33, 1497–1537.

Bartlett, P. L., N. Harvey, C. Liaw, and A. Mehrabian (2017): "Nearly-tight VC-dimension bounds for piecewise linear neural networks," in *Proceedings of the 22nd Annual Conference on Learning Theory (COLT 2017)*.

Bauer, B. and M. Kohler (2019): "On deep learning as a remedy for the curse of dimensionality in nonparametric regression," *Annals of Statistics*, 47, 2261–2285.

Belloni, A., V. Chernozhukov, I. Fernández-Val, and C. Hansen (2017): "Program Evaluation and Causal Inference With High-Dimensional Data," *Econometrica*, 85, 233–298.

Belloni, A., V. Chernozhukov, and C. Hansen (2014): "Inference on Treatment Effects after Selection Amongst High-Dimensional Controls," *Review of Economic Studies*, 81, 608–650.

Bickel, P. J. and Y. Ritov (1988): "Estimating Integrated Squared Density Derivatives: Sharp Best Order of Convergence Estimates," *Sankhyā*, 50, 381–393.

Bickel, P. J., Y. Ritov, and A. B. Tsybakov (2009): "Simultaneous Analysis of LASSO and Dantzig Selector," *The Annals of Statistics*, 37, 1705–1732.

Cattaneo, M. D., M. Jansson, and X. Ma (2018): "Two-step Estimation and Inference with Possibly Many Included Covariates," *arXiv:1807.10100, Review of Economic Studies*, forthcoming.

Chen, X., H. Hong, and A. Tarozzi (2008): "Semiparametric Efficiency in GMM Models With Auxiliary Data," *The Annals of Statistics*, 36, 808–843.

Chen, X. and H. White (1999): "Improved rates and asymptotic normality for nonparametric neural network estimators," *IEEE Transactions on Information Theory*, 45, 682–691.

Chernozhukov, V., D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins (2018): "Double/debiased machine learning for treatment and structural parameters," *The Econometrics Journal*, 21, C1–C68.

Daniely, A. (2017): "Depth separation for neural networks," *arXiv preprint arXiv:1702.08489*.

Dou, X. and T. Liang (2020): "Training Neural Networks as Learning Data-Adaptive Kernels: Provable Representation and Approximation Benefits," *Journal of the American Statistical Association*, 0, 1–14.

Farrell, M. H. (2015): "Robust Inference on Average Treatment Effects with Possibly More Covariates than Observations," *arXiv:1309.4686, Journal of Econometrics*, 189, 1–23.

Farrell, M. H., T. Liang, and S. Misra (2019a): "Deep Neural Networks for Estimation and Inference," *arXiv:1809.09953*.

——— (2019b): "Supplement to 'Deep Neural Networks for Estimation and Inference'," *Supplemental Material*.

Ghorbani, B., S. Mei, T. Misiakiewicz, and A. Montanari (2019): "Discussion of 'Nonparametric regression using deep neural networks with ReLU activation function'," *Annals of Statistics*, forthcoming.

Gine, E. and R. Nickl (2016): *Mathematical Foundations of Infinite-Dimensional Models*, Cambridge.

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

Hahn, J. (1998): "On the Role of the Propensity Score in Efficient Semiparametric Estimation of Average Treatment Effects," *Econometrica*, 66, 315–331.

Hanin, B. (2017): "Universal function approximation by deep neural nets with bounded width and relu activations," *arXiv preprint arXiv:1708.02691*.

Hartford, J., G. Lewis, K. Leyton-Brown, and M. Taddy (2017): "Deep iv: A flexible approach for counterfactual prediction," in *International Conference on Machine Learning*, 1414–1423.

Hitsch, G. J. and S. Misra (2018): "Heterogeneous Treatment Effects and Optimal Targeting Policy Evaluation," *SSRN preprint 3111957*.

Javanmard, A. and A. Montanari (2014): "Confidence intervals and hypothesis testing for high-dimensional regression," *The Journal of Machine Learning Research*, 15, 2869–2909.

Kingma, D. P. and J. Ba (2014): "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*.

Koltchinskii, V. (2006): "Local Rademacher complexities and oracle inequalities in risk minimization," *The Annals of Statistics*, 34, 2593–2656.

——— (2011): *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems*, Springer-Verlag.

Koltchinskii, V. and D. Panchenko (2000): "Rademacher processes and bounding the risk of function learning," in *High dimensional probability II*, Springer, 443–457.

Kutyniok, G. (2019): "Discussion of 'Nonparametric regression using deep neural networks with ReLU activation function'," *Annals of Statistics*, forthcoming.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner (1998): "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 86, 2278–2324.

Liang, T. (2018): "On How Well Generative Adversarial Networks Learn Densities: Nonparametric and Parametric Results," *arXiv:1811.03179*.

Liang, T., A. Rakhlin, and K. Sridharan (2015): "Learning with square loss: Localization through offset Rademacher complexity," in *Conference on Learning Theory*, 1260–1285.

Ma, X. and J. Wang (2018): "Robust Inference Using Inverse Probability Weighting," *arXiv preprint arXiv:1810.11397*.

Mendelson, S. (2003): "A few notes on statistical learning theory," in *Advanced lectures on machine learning*, Springer, 1–40.

——— (2014): "Learning without concentration," in *Conference on Learning Theory*, 25–39.

Mhaskar, H. N. and T. Poggio (2016): "Deep vs. shallow networks: An approximation theory perspective," *Analysis and Applications*, 14, 829–848.

Nair, V. and G. E. Hinton (2010): "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 807–814.

Polson, N. G. and V. Ročková (2018): "Posterior concentration for sparse deep learning," in *Advances in Neural Information Processing Systems*, 930–941.

Robins, J., E. T. Tchetgen, L. Li, and A. van der Vaart (2009): "Semiparametric Minimax Rates," *Electronic Journal of Statistics*, 3, 1305–1321.

Schmidt-Hieber, J. (2019): "Nonparametric regression using deep neural networks with ReLU activation function," *arXiv:1708.06633, Annals of Statistics*, forthcoming.

Shalit, U., F. D. Johansson, and D. Sontag (2017): "Estimating individual treatment effect: generalization bounds and algorithms," *arXiv preprint arXiv:1606.03976*.

Shamir, O. (2019): "Discussion of 'Nonparametric regression using deep neural networks with ReLU activation function'," *Annals of Statistics*, forthcoming.

Stone, C. J. (1982): "Optimal global rates of convergence for nonparametric regression," *The annals of statistics*, 1040–1053.

Tan, Z. (2018): "Model-assisted inference for treatment effects using regularized calibrated estimation with high-dimensional data," *arXiv preprint arXiv:1801.09817*.

Telgarsky, M. (2016): "Benefits of depth in neural networks," *arXiv preprint arXiv:1602.04485*.

van de Geer, S., P. Buhlmann, Y. Ritov, and R. Dezeure (2014): "On Asymptotically Optimal Confidence Regions and Tests for High-Dimensional Models," *The Annals of Statistics*, 42, 1166–1202.

WAGER, S. AND S. ATHEY (2018): "Estimation and Inference of Heterogeneous Treatment Effects using Random Forests," *Journal of the American Statistical Association,* forthcoming.

WESTREICH, D., J. LESSLER, AND M. J. FUNK (2010): "Propensity score estimation: neural networks, support vector machines, decision trees (CART), and meta-classifiers as alternatives to logistic regression," *Journal of clinical epidemiology*, 63, 826–833.

WHITE, H. (1992): *Artificial neural networks: approximation and learning theory*, Blackwell Publishers, Inc.

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

——— (2018): "Optimal approximation of continuous functions by very deep ReLU networks," *arXiv preprint arXiv:1802.03620*.

ZHANG, C., S. BENGIO, M. HARDT, B. RECHT, AND O. VINYALS (2016): "Understanding deep learning requires rethinking generalization," *arXiv preprint arXiv:1611.03530*.

TABLE I

Deep Network Architectures

DNN	Learning Rate	Widths $[H_1, H_2, ...]$	Dropout $[H_1, H_2, ...]$	Total Parameters	Validation Loss	Training Loss	$\mathbb{P}_n[\widehat{\tau}(\boldsymbol{x}_i) < 0]$
1	0.0003	[60]	[0.5]	8702	1405.62	1748.91	0.0014
2	0.0003	[100]	[0.5]	14502	1406.48	1751.87	0.0251
3	0.0001	[30, 20]	[0.5, 0]	4952	1408.22	1751.20	0.0072
4	0.0009	[30, 10]	[0.3, 0.1]	4622	1408.56	1751.62	0.0138
5	0.0003	[30, 30]	[0, 0]	5282	1403.57	1738.59	0.0226
6	0.0003	[30, 30]	[0.5, 0]	5282	1408.57	1755.28	0.0066
7	0.0003	[100, 30, 20]	[0.5, 0.5, 0]	17992	1408.62	1751.52	0.0103
8	0.00005	[80, 30, 20]	[0.5, 0.5, 0]	14532	1413.70	1756.93	0.0002

Notes: All networks use the ReLU activation function. The width of each layer is shown, e.g. Architecture 3 consists of two layers, with 30 and 20 hidden units respectively. The final column shows the portion of estimated individual treatment effects below zero.

TABLE II

Average Treatment Effect Estimates and Counterfactual Profits from Three Targeting Strategies, with 95% Confidence Intervals

	Average Effect		**Never Treat**		**Blanket Treatment**		**Loyalty Policy**	
DNN	$\widehat{\tau}$	95% CI	$\widehat{\pi}(s)$	95% CI	$\widehat{\pi}(s)$	95% CI	$\widehat{\pi}(s)$	95% CI
1	2.606	[2.273 , 2.932]	2.016	[1.923 , 2.110]	2.234	[2.162 , 2.306]	2.367	[2.292 , 2.443]
2	2.577	[2.252 , 2.901]	2.022	[1.929 , 2.114]	2.229	[2.157 , 2.301]	2.363	[2.288 , 2.438]
3	2.547	[2.223 , 2.872]	2.027	[1.934 , 2.120]	2.224	[2.152 , 2.296]	2.358	[2.283 , 2.434]
4	2.488	[2.160 , 2.817]	2.037	[1.944 , 2.130]	2.213	[2.140 , 2.286]	2.350	[2.274 , 2.425]
5	2.459	[2.127 , 2.791]	2.043	[1.950 , 2.136]	2.208	[2.135 , 2.281]	2.345	[2.269 , 2.422]
6	2.430	[2.093 , 2.767]	2.048	[1.954 , 2.142]	2.202	[2.128 , 2.277]	2.341	[2.263 , 2.418]
7	2.400	[2.057 , 2.744]	2.053	[1.959 , 2.148]	2.197	[2.122 , 2.272]	2.336	[2.258 , 2.414]
8	2.371	[2.021 , 2.721]	2.059	[1.963 , 2.154]	2.192	[2.116 , 2.268]	2.332	[2.253 , 2.411]

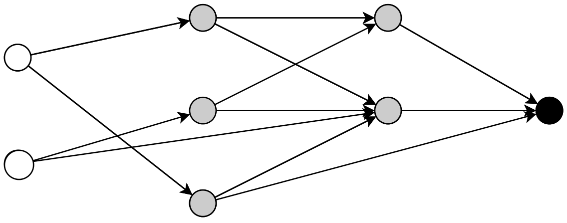

Figure 1: Illustration of a feedforward neural network with $W = 18$, $L = 2$, $U = 5$, and input dimension $d = 2$. The input units are shown in white at left, the output in black at right, and the hidden units in grey between them.

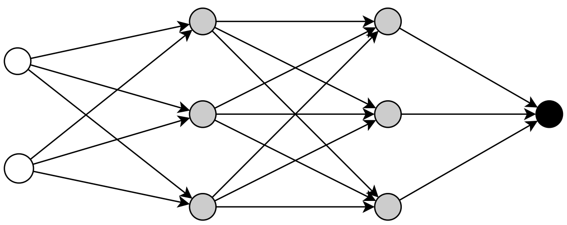

Figure 2: Illustration of multi-layer perceptron $\mathcal{F}_{\text{MLP}}$ with $H = 3$, $L = 2$ ($U = 6$, $W = 25$), and input dimension $d = 2$.

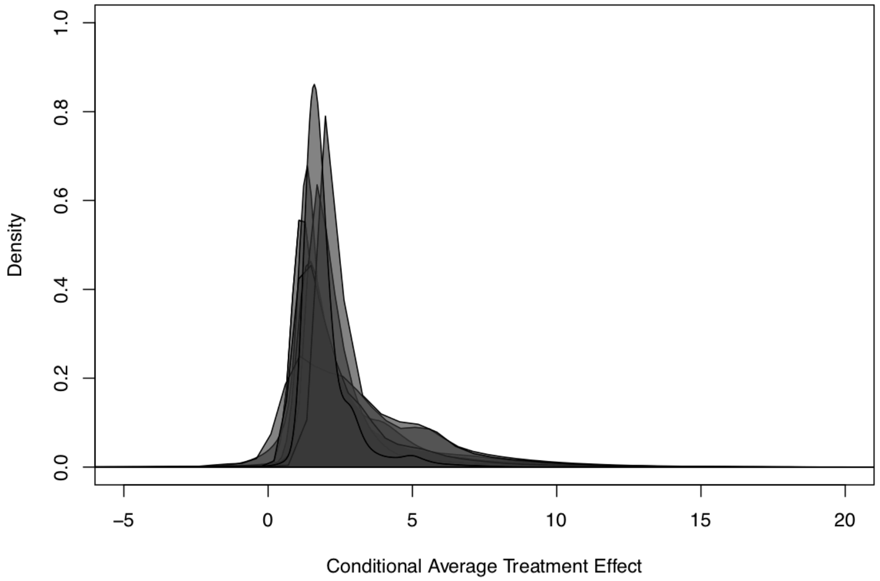

Figure 3: Conditional Average Treatment Effects Across Architectures

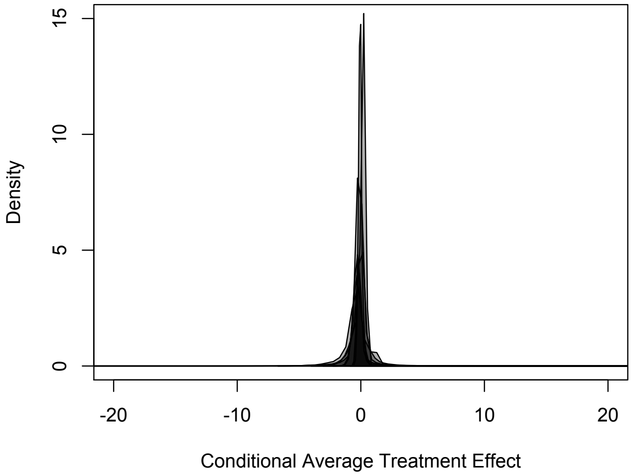

Figure 4: Placebo Test

A. Proofs

In this section we provide a proof of Theorems 1 and 2, our main theoretical results for deep ReLU networks, and their corollaries. The proof proceeds in several steps. We first give the main breakdown and bound the bias (approximation error) term. We then turn our attention to the empirical process term, to which we apply our localization. Much of the proof uses a generic architecture, and thus pertains to both results. We will specialize the architecture to the multi-layer perceptron only when needed later on. Other special cases and related results are covered in Section A.4. Supporting Lemmas are stated in Section B.

The statements of Theorems 1 and 2 assume that n is large enough. Precisely, we require $n > (2eM)^2 \vee \text{Pdim}(\mathcal{F}_{\text{DNN}})$. For notational simplicity we will denote $\widehat{f}_{\text{DNN}} := \widehat{f}$, see (2.4), and $\epsilon_{\text{DNN}} := \epsilon_n$, see Assumption 3. As we are simultaneously consider Theorems 1 and 2, the generic notation DNN will be used throughout.

A.1. Main Decomposition and Bias Term

Referring to Assumption 3, define the best approximation realized by the deep ReLU network class $\mathcal{F}_{\text{DNN}}$ as

$$f_n := \underset{\substack{f \in \mathcal{F}_{\text{DNN}} \\ \|f\|_\infty \leq 2M}}{\arg\min} \|f - f_*\|_\infty.$$

By definition, $\epsilon_n := \epsilon_{\text{DNN}} := \|f_n - f_*\|_\infty$.

Recalling the optimality of the estimator in (2.4), we know, as both f_n and $\widehat{f}$ are in $\mathcal{F}_{\text{DNN}}$, that

$$-\mathbb{E}_n[\ell(\widehat{f}, \boldsymbol{z})] + \mathbb{E}_n[\ell(f_n, \boldsymbol{z})] \geq 0.$$

This result does not hold for f_* in place of f_n, because $f_* \notin \mathcal{F}_{\text{DNN}}$. Using the above display and the curvature of Equation (2.1) (which does not hold with f_n in place of f_* therein), we obtain

$$\begin{aligned}
&c_1\|\widehat{f} - f_*\|^2_{L_2(\boldsymbol{X})} \leq \mathbb{E}[\ell(\widehat{f}, \boldsymbol{z})] - \mathbb{E}[\ell(f_*, \boldsymbol{z})] \\
\leq& \mathbb{E}[\ell(\widehat{f}, \boldsymbol{z})] - \mathbb{E}[\ell(f_*, \boldsymbol{z})] - \mathbb{E}_n[\ell(\widehat{f}, \boldsymbol{z})] + \mathbb{E}_n[\ell(f_n, \boldsymbol{z})] \\
=& \mathbb{E}\left[\ell(\widehat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] - \mathbb{E}_n\left[\ell(\widehat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] + \mathbb{E}_n\left[\ell(f_n, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] \\
=& \left(\mathbb{E} - \mathbb{E}_n\right)\left[\ell(\widehat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] + \mathbb{E}_n\left[\ell(f_n, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right]. \qquad \text{(A.1)}
\end{aligned}$$

Equation (A.1) is the main decomposition that begins the proof. The decomposition must be done this way because of the above notes regarding f_* and f_n. The first term is the empirical process term that will be treated in the subsequent subsection. For the second term in (A.1), the bias term or approximation error, we apply Bernstein's inequality to find that, with probability at least $1 - e^{-\tilde{\gamma}}$,

$$\begin{aligned}
&\mathbb{E}_n\left[\ell(f_n, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] \leq \mathbb{E}\left[\ell(f_n, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] + \sqrt{\frac{2C_\ell^2\|f_n - f_*\|_\infty^2\tilde{\gamma}}{n}} + \frac{21C_\ell M\tilde{\gamma}}{3n} \\
\leq& c_2\mathbb{E}\left[\|f_n - f_*\|^2\right] + \sqrt{\frac{2C_\ell^2\|f_n - f_*\|_\infty^2\tilde{\gamma}}{n}} + \frac{7C_\ell M\tilde{\gamma}}{n} \\
\leq& c_2\epsilon_n^2 + \epsilon_n\sqrt{\frac{2C_\ell^2\tilde{\gamma}}{n}} + \frac{7C_\ell M\tilde{\gamma}}{n}, \qquad \text{(A.2)}
\end{aligned}$$

using the Lipschitz and curvature of the loss function defined in Equation (2.1) and $\mathbb{E}\left[\|f_n - f_*\|^2\right] \leq \|f_n - f_*\|_\infty^2$, along with the definition of ϵ_n^2.

Once the empirical process term is controlled (in Section A.2), the two bounds will be brought back together to compute the final result, see Section A.3.

A.2. Localization Analysis

We now turn to bounding the first term in (A.1) (the empirical processes term) using a localized analysis that derives bounds based on scale insensitive complexity measure. The ideas of our localization are rooted in Koltchinskii and Panchenko (2000) and Bartlett et al. (2005), and related to Koltchinskii (2011). Localization analysis extending to the unbounded f case has been developed in Liang et al. (2015); Mendelson (2014). This proof section proceeds in several steps.

A key quantity is the Rademacher complexity of the function class at hand. Given i.i.d. Rademacher draws, $\eta_i = \pm 1$ with equal probability independent of the data, the random variable $R_n\mathcal{F}$, for a function class $\mathcal{F}$, is defined as

$$R_n\mathcal{F} := \sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \eta_i f(\boldsymbol{x}_i).$$

Intuitively, $R_n\mathcal{F}$ measures how flexible the function class is for predicting random signs. Taking the expectation of $R_n\mathcal{F}$ conditioned on the data we obtain the *empirical Rademacher complexity*, denoted $\mathbb{E}_\eta[R_n\mathcal{F}]$. When the expectation is taken over both the data and the draws η_i, $\mathbb{E}R_n\mathcal{F}$, we get the *Rademacher complexity*.

A.2.1. Step I: Quadratic Process

The first step is to show that, with high probability, the empirical L_2 norm of the error $(f - f_*)$ is at most twice the population L_2 norm bound for the same error, for certain functions f outside a certain critical radius. This will be an ingredient to be used later on. To do so, we study the quadratic process

$$\|f - f_*\|_n^2 - \|f - f_*\|_{L_2(\boldsymbol{X})}^2 = \mathbb{E}_n(f - f_*)^2 - \mathbb{E}(f - f_*)^2.$$

We will apply the symmetrization of Lemma 5 to $g = (f - f_*)^2$ restricted to a radius $\|f - f_*\|_{L_2(\boldsymbol{X})} \leq r$. This function g has variance bounded as

$$\mathbb{V}[g] \leq \mathbb{E}[g^2] \leq \mathbb{E}((f - f_*)^4) \leq 9M^2r^2.$$

Writing $g = (f + f_*)(f - f_*)$, we see that by Assumption 1, $|g| \leq 3M|f - f_*| \leq 9M^2$, where the first inequality verifies that g has a Lipschitz constant of $3M$ (when viewed as a function of its argument f), and second that g itself is bounded. We therefore apply Lemma 5, to obtain, with probability at least $1 - \exp(-\tilde{\gamma})$, that for any $f \in \mathcal{F}$ with $\|f - f_*\|_{L_2(\boldsymbol{X})} \leq r$,

$$\begin{aligned}&\mathbb{E}_n(f - f_*)^2 - \mathbb{E}(f - f_*)^2\\ &\leq 3\mathbb{E}R_n\{g = (f - f_*)^2 : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq r\} + 3Mr\sqrt{\frac{2\tilde{\gamma}}{n}} + \frac{36M^2}{3}\frac{\tilde{\gamma}}{n}\end{aligned}$$

$$\leq 18M\mathbb{E}R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq r\} + 3Mr\sqrt{\frac{2\tilde{\gamma}}{n}} + \frac{12M^2\tilde{\gamma}}{n}, \tag{A.3}$$

where the second inequality applies Lemma 2 to the Lipschitz functions $\{g\}$ (as a function of the real values $f(\boldsymbol{x})$) and iterated expectations.

Suppose the radius r satisfies

$$r^2 \geq 18M\mathbb{E}R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq r\} \tag{A.4}$$

and

$$r^2 \geq \frac{6\sqrt{6}M^2\tilde{\gamma}}{n}. \tag{A.5}$$

Then we conclude from from (A.3) that

$$\mathbb{E}_n(f - f_*)^2 \leq r^2 + r^2 + 3Mr\sqrt{\frac{2\tilde{\gamma}}{n}} + \frac{12M^2\tilde{\gamma}}{n} \leq (2r)^2 \tag{A.6}$$

where the first inequality uses (A.4) and the second line uses (A.5). This means that for r above the "critical radius" (see **Step III**), the empirical L_2-norm is at most twice the population one with probability at least $1 - \exp(-\tilde{\gamma})$.

A.2.2. Step II: One Step Improvement

In this step we will show that given a bound on $\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})}$ we can use this bound as information to obtain a tighter bound, if the initial bound is loose as made precise at the end of this step. This tightening will then be pursued to its limit in **Step III**, which leads to the final rate obtained in **Step IV**. **Step I** will be used herein.

Suppose we know that for some r_0, $\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq r_0$. We may always start with $r_0 = 3M$ given Assumption 1 and (2.4). Apply Lemma 5 with $\mathcal{G} := \{g = \ell(f, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z}) : f \in \mathcal{F}_{\mathrm{DNN}}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq r_0\}$, we find that, with probability at least $1 - 2e^{-\tilde{\gamma}}$, the empirical process term of (A.1) is bounded as

$$(\mathbb{E} - \mathbb{E}_n)\left[\ell(\widehat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] \leq 6\mathbb{E}_\eta R_n\mathcal{G} + \sqrt{\frac{2C_\ell^2 r_0^2\tilde{\gamma}}{n}} + \frac{23 \cdot 3MC_\ell}{3}\frac{\tilde{\gamma}}{n}, \tag{A.7}$$

where the middle term is due to the following variance calculation (recall Equation (2.1))

$$\mathbb{V}[g] \leq \mathbb{E}[g^2] = \mathbb{E}[|\ell(f, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})|^2] \leq C_\ell^2\mathbb{E}(f - f_*)^2 \leq C_\ell^2 r_0^2$$

Here the fact that Lemma 5 is variance dependent, and that the variance depends on the radius r_0, is important. It is this property which enables a sharpening of the rate with step-by-step reductions in the variance bound, as in Section A.2.4.

For the empirical Rademacher complexity term, the first term of (A.7), Lemma 2, **Step I**, and Lemma 3 (notation defined there), yield

$$\begin{aligned}\mathbb{E}_\eta R_n\mathcal{G} &= \mathbb{E}_\eta R_n\{g : g = \ell(f, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z}), f \in \mathcal{F}_{\mathrm{DNN}}, \|f - f_*\| \leq r_0\} \\ &\leq 2C_\ell\mathbb{E}_\eta R_n\{f - f_* : f \in \mathcal{F}_{\mathrm{DNN}}, \|f - f_*\| \leq r_0\}\end{aligned}$$

$$
\begin{aligned}
&\leq 2C_\ell \mathbb{E}_\eta R_n\{f - f_* : f \in \mathcal{F}_{\mathrm{DNN}}, \|f - f_*\|_n \leq 2r_0\}\\
&\leq 2C_\ell \inf_{0<\alpha<2r_0} \left\{4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^{2r_0} \sqrt{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{DNN}}, \|\cdot\|_n)} d\delta\right\}\\
&\leq 2C_\ell \inf_{0<\alpha<2r_0} \left\{4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^{2r_0} \sqrt{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{DNN}}|_{\boldsymbol{x}_1,\dots,\boldsymbol{x}_n}, \infty)} d\delta\right\} \quad ,
\end{aligned}
$$

with probability $1 - \exp(-\tilde{\gamma})$ (when applying **Step I**). Recall Lemma 4, one can further upper bound the entropy integral when $n > \mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})$,

$$
\begin{aligned}
&\inf_{0<\alpha<2r_0} \left\{4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^{2r_0} \sqrt{\log \mathcal{N}(\delta, \mathcal{F}_{\mathrm{DNN}}|_{x_1,\dots,x_n}, \infty)} d\delta\right\}\\
&\leq \inf_{0<\alpha<2r_0} \left\{4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^{2r_0} \sqrt{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}}) \log \frac{2eMn}{\delta \cdot \mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}} d\delta\right\}\\
&\leq 32 r_0 \sqrt{\frac{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}{n}\left(\log \frac{2eM}{r_0} + \frac{3}{2}\log n\right)}
\end{aligned}
$$

with a particular choice of $\alpha = 2r_0\sqrt{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})/n} < 2r_0$. Therefore, whenever $r_0 \geq 1/n$ and $n \geq (2eM)^2$,

$$
\mathbb{E}_\eta R_n \mathcal{G} \leq 128 C_\ell r_0 \sqrt{\frac{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}{n} \log n}.
$$

Applying this bound to (A.7), we have

$$
(\mathbb{E} - \mathbb{E}_n)\left[\ell(\widehat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] \leq K r_0 \sqrt{\frac{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}{n} \log n} + r_0\sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}} + \frac{23 M C_\ell \tilde{\gamma}}{n} \tag{A.8}
$$

where $K = 6 \times 128 C_\ell$.

Going back now to the main decomposition, plug (A.8) and (A.2) into (A.1), and we overall have found that, with probability at least $1 - 4\exp(-\tilde{\gamma})$, the following holds:

$$
\begin{aligned}
&c_1 \|\widehat{f} - f_*\|^2_{L_2(\boldsymbol{X})}\\
&\leq K r_0 \sqrt{\frac{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}{n} \log n} + r_0 \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}} + \frac{23 M C_\ell \tilde{\gamma}}{n} + \left(c_2 \epsilon_n^2 + \epsilon_n \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}} + \frac{7 C_\ell M \tilde{\gamma}}{n}\right)\\
&\leq r_0 \cdot \left(K \sqrt{\frac{\mathrm{Pdim}(\mathcal{F}_{\mathrm{DNN}})}{n} \log n} + \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}}\right) + c_2 \epsilon_n^2 + \epsilon_n \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}} + 30 M C_\ell \frac{\tilde{\gamma}}{n}\\
&\leq r_0 \cdot \left(K\sqrt{C} \sqrt{\frac{WL\log W}{n} \log n} + \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}}\right) + c_2 \epsilon_n^2 + \epsilon_n \sqrt{\frac{2C_\ell^2 \tilde{\gamma}}{n}} + 30 M C_\ell \frac{\tilde{\gamma}}{n}. \qquad \text{(A.9)}
\end{aligned}
$$

The last line applies Lemma 6. Therefore, whenever $\epsilon_n \ll r_0$ and $\sqrt{\frac{WL\log W}{n} \log n} \ll r_0$, the knowledge that $\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq r_0$ implies that (with high probability) $\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq r_1$, for $r_1 \ll r_0$. One can recursively improve the bound r to a fixed point/radius r_*, which describes the fundamental difficulty of the problem. This is done in the course of the next two steps.

A.2.3. Step III: Critical Radius

We now use the tightening of **Step II** to obtain the critical radius for this problem that is then used as an input in the final rate derivation of **Step IV**. Formally, define the critical radius r_* to be the largest fixed point

$$r_* = \inf \left\{ r > 0 : 18M\mathbb{E}R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq s\} < s^2, \forall s \geq r \right\}.$$

By construction this obeys (A.4), and thus so does $2r_*$. Denote the event E (depending on the data) to be

$$E = \left\{ \|f - f_*\|_n \leq 4r_*, \quad \text{for all } f \in \mathcal{F} \text{ and } \quad \|f - f_*\|_{L_2(\boldsymbol{X})} \leq 2r_* \right\}$$

and $\mathbb{1}_E$ to be the indicator that event E holds. We know from (A.6) that $\mathbb{P}(\mathbb{1}_E = 1) \geq 1 - n^{-1}$, provided $r_* \geq \sqrt{18}M\sqrt{\log n/n}$ to satisfy (A.5).

We can now give an upper bound for the the critical radius r_*. Using the logic of **Step II** to bound the empirical Rademacher complexity, and then applying Lemma 6, we find that

$$\begin{aligned}
r_*^2 &\leq 18M\mathbb{E}R_n\big\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq r_*\big\} \\
&\leq 18M\mathbb{E}R_n\big\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq 2r_*\big\} \\
&\leq 18M\mathbb{E}\big\{\mathbb{E}_\eta R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_n \leq 4r_*\}\mathbb{1}_E + 3M(1 - \mathbb{1}_E)\big\} \\
&\leq 36MK\sqrt{C} \cdot r_*\sqrt{\frac{WL\log W}{n}\log n} + 36M^2\frac{1}{n} \\
&\leq 72MK\sqrt{C} \cdot r_*\sqrt{\frac{WL\log W}{n}\log n},
\end{aligned}$$

with the last line relying on the above restriction that $r_* \geq \sqrt{18}M\sqrt{\log n/n}$. Dividing through by r_* yields the final bound:

$$r_* \leq 72MK\sqrt{C}\sqrt{\frac{WL\log W}{n}\log n}. \tag{A.10}$$

A.2.4. Step IV: Localization

We are now able to derive the final rate using a localization argument. This applies the results of **Step I** and **Step II** repeatedly. Divide the space $\mathcal{F}_{\text{DNN}}$ into shells of increasing radius by intersecting it with the L_2 balls

$$B(f_*, \bar{r}), B(f_*, 2\bar{r})\backslash B(f_*, \bar{r}), \ldots B(f_*, 2^l\bar{r})\backslash B(f_*, 2^{l-1}\bar{r}) \tag{A.11}$$

where $l \geq 1$ is chosen to be the largest integer no greater than $\log_2 \frac{2M}{\sqrt{(\log n)/n}}$. We will proceed to find a bound on $\bar{r}$ which determines the final rate results.

Suppose $\bar{r} > r_*$. Then for each shell, **Step I** and the union bound imply that with probability at least $1 - 2l\exp(-\tilde{\gamma})$,

$$\|f - f_*\|_{L_2(\boldsymbol{X})} \leq 2^j\bar{r} \ \Rightarrow \ \|f - f_*\|_n \leq 2^{j+1}\bar{r}. \tag{A.12}$$

Further, suppose that for some $j \leq l$

$$\widehat{f} \in B(f_*, 2^j \bar{r}) \backslash B(f_*, 2^{j-1}\bar{r}). \tag{A.13}$$

Then applying the one step improvement argument in **Step II** (again the variance dependence captured in Lemma 5 is crucial, here reflected in the variance within each shell), Equation (A.9) yields that with probability at least $1 - 4\exp(-\tilde{\gamma})$,

$$\begin{aligned}\|\widehat{f} - f_*\|^2_{L_2(\boldsymbol{X})} &\leq \frac{1}{c_1}\left\{ 2^j \bar{r} \cdot \left(K\sqrt{C}\sqrt{\frac{WL\log W}{n}}\log n + \sqrt{\frac{2C_\ell^2 t}{n}} \right) + c_2\epsilon_n^2 + \epsilon_n \sqrt{\frac{2C_\ell^2\tilde{\gamma}}{n}} + 30MC_\ell\frac{\tilde{\gamma}}{n} \right\} \\ &\leq 2^{2j-2}\bar{r}^2,\end{aligned}$$

if the following two conditions hold:

$$\begin{aligned}&\frac{1}{c_1}\left(K\sqrt{C}\sqrt{\frac{WL\log W}{n}}\log n + \sqrt{\frac{2C_\ell^2\tilde{\gamma}}{n}} \right) \leq \frac{1}{2}2^j\bar{r} \\ &\quad\frac{1}{c_1}\left(c_2\epsilon_n^2 + \epsilon_n\sqrt{\frac{2C_\ell^2\tilde{\gamma}}{n}} + 26MC_\ell\frac{\tilde{\gamma}}{n} \right) \leq \frac{1}{4}2^{2j}\bar{r}^2.\end{aligned}$$

It is easy to see that these two hold for all j if we choose

$$\bar{r} = \frac{8}{c_1}\left(K\sqrt{C}\sqrt{\frac{WL\log W}{n}}\log n + \sqrt{\frac{2C_\ell^2\tilde{\gamma}}{n}} \right) + \left(\sqrt{\frac{2(c_2 \vee 1)}{c_1}}\epsilon_n + \sqrt{\frac{120MC_\ell}{c_1}\frac{\tilde{\gamma}}{n}} \right) + r_*. \tag{A.14}$$

Therefore with probability at least $1 - 6l\exp(-\tilde{\gamma})$, we can perform shell-by-shell argument combining the results in **Step I** and **Step II**:

$$\begin{aligned}&\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq 2^l\bar{r} \ \text{ and } \ \|\widehat{f} - f_*\|_n \leq 2^{l+1}\bar{r} \\ &\quad\text{implies } \ \|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq 2^{l-1}\bar{r} \ \text{ and } \ \|\widehat{f} - f_*\|_n \leq 2^l\bar{r} \\ &\qquad\ldots\ldots \\ &\qquad\quad\text{implies } \ \|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq 2^0\bar{r} \ \text{ and } \ \|\widehat{f} - f_*\|_n \leq 2^1\bar{r}.\end{aligned}$$

The "and" part of each line follows from **Step I** and the implication uses the above argument following **Step II**. Therefore in the end, we conclude with probability at least $1 - 6l\exp(-\tilde{\gamma})$,

$$\|\widehat{f} - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r} \ , \tag{A.15}$$

$$\|\widehat{f} - f_*\|_n \leq 2\bar{r} \ . \tag{A.16}$$

Therefore choose $\gamma = -\log(6l) + \tilde{\gamma}$, we know from (A.14), and the upper bound on r_* in (A.10)

$$\begin{aligned}\bar{r} \leq \frac{8}{c_1}&\left(K\sqrt{C}\sqrt{\frac{WL\log W}{n}}\log n + \sqrt{\frac{2C_\ell^2(\log\log n + \gamma)}{n}} \right) \\ &\qquad + \left(\sqrt{\frac{2(c_2 \vee 1)}{c_2}}\epsilon_n + \sqrt{\frac{120MC_\ell}{c_1}\frac{\log\log n + \gamma}{n}} \right) + r_*\end{aligned}$$

$$\leq C'\left(\sqrt{\frac{WL\log W}{n}}\log n + \sqrt{\frac{\log\log n + \gamma}{n}} + \epsilon_n\right), \tag{A.17}$$

with some constant $C' > 0$ that does not depend on n. This completes the proof of Theorem 2.

A.3. Final Steps for the MLP case

For the multi-layer perceptron, $W \leq C \cdot H^2 L$, and plugging this into the bound (A.17), we obtain

$$C'\left(\sqrt{\frac{H^2L^2\log(H^2L)}{n}}\log n + \sqrt{\frac{\log\log n + \gamma}{n}} + \epsilon_n\right)$$

To optimize this upper bound on $\bar{r}$, we need to specify the trade-offs in ϵ_n and H and L. To do so, we utilize the MLP-specific approximation rate of Lemma 7 and the embedding of Lemma 1. Lemma 1 implies that, for any ϵ_n, one can embed the approximation class $\mathcal{F}_{\mathrm{DNN}}$ given by Lemma 7 into a standard MLP architecture $\mathcal{F}_{\mathrm{MLP}}$, where specifically

$$\begin{aligned} H &= H(\epsilon_n) \leq W(\epsilon_n)L(\epsilon_n) \leq C^2\epsilon_n^{-\frac{d}{\beta}}(\log(1/\epsilon_n)+1)^2, \\ L &= L(\epsilon_n) \leq C \cdot (\log(1/\epsilon_n)+1). \end{aligned}$$

For standard MLP architecture $\mathcal{F}_{\mathrm{MLP}}$,

$$H^2L^2\log(H^2L) \leq \tilde{C} \cdot \epsilon_n^{-\frac{2d}{\beta}}(\log(1/\epsilon_n)+1)^7.$$

Thus we can optimize the upper bound

$$\bar{r} \leq C'\left(\sqrt{\frac{\epsilon_n^{-\frac{2d}{\beta}}(\log(1/\epsilon_n)+1)^7}{n}}\log n + \sqrt{\frac{\log\log n + \gamma}{n}} + \epsilon_n\right)$$

by choosing $\epsilon_n = n^{-\frac{\beta}{2(\beta+d)}}$, $H \asymp \cdot n^{\frac{d}{2(\beta+d)}}\log^2 n$, $L \asymp \cdot \log n$. This gives

$$\bar{r} \leq C\left(n^{-\frac{\beta}{2(\beta+d)}}\log^4 n + \sqrt{\frac{\log\log n + t'}{n}}\right).$$

Hence putting everything together, with probability at least $1-\exp(-\gamma)$,

$$\begin{aligned} \mathbb{E}(\widehat{f} - f_*)^2 &\leq \bar{r}^2 \leq C\left(n^{-\frac{\beta}{\beta+d}}\log^8 n + \frac{\log\log n + \gamma}{n}\right), \\ \mathbb{E}_n(\widehat{f} - f_*)^2 &\leq (2\bar{r})^2 \leq 4C\left(n^{-\frac{\beta}{\beta+d}}\log^8 n + \frac{\log\log n + \gamma}{n}\right). \end{aligned}$$

This completes the proof of Theorem 1.

A.4. Proof of Corollaries 1 and 2

For Corollary 1, we want to optimize

$$\frac{WL\log U}{n}\log n + \frac{\log\log n + \gamma}{n} + \epsilon_{\mathrm{DNN}}^2.$$

Yarotsky (2017, Theorem 1) shows that for the approximation error ϵ_{DNN} to obey $\epsilon_{\mathrm{DNN}} \leq \epsilon$, it suffices to choose $W, U \propto \epsilon^{-\frac{d}{\beta}}(\log(1/\epsilon)+1)$ and $L \propto (\log(1/\epsilon)+1)$, given the specific architecture described therein. Therefore, we attain $\epsilon \asymp n^{-\beta/(2\beta+d)}$ by setting $W, U \asymp n^{d/(2\beta+d)}$ and $L \asymp \log n$, yielding the desired result.

For Corollary 2, we need to optimize

$$\frac{H^2 L_2 \log(HL)}{n}\log n + \frac{\log\log n + \gamma}{n} + \epsilon_{\mathrm{MLP}}^2.$$

Yarotsky (2018, Theorem 1) shows that for the approximation error ϵ_{MLP} to obey $\epsilon_{\mathrm{MLP}} \leq \epsilon$, it suffices to choose $H \propto 2d + 10$ and $L \propto \epsilon^{-\frac{d}{2}}$, given the specific architecture described therein. Thus, for $\epsilon \asymp n^{-1/(2+d)}$ we take $L \asymp n^{-d/(4+2d)}$, and the result follows.

B. Supporting Lemmas

First, we show that one can embed a feedforward network into the multi-layer perceptron architecture by adding auxiliary hidden nodes. This idea is due to Yarotsky (2018).

Lemma 1 (Embedding) *For any function $f \in \mathcal{F}_{\mathrm{DNN}}$, there is a $g \in \mathcal{F}_{\mathrm{MLP}}$, with $H \leq WL + U$, such that $g = f$.*

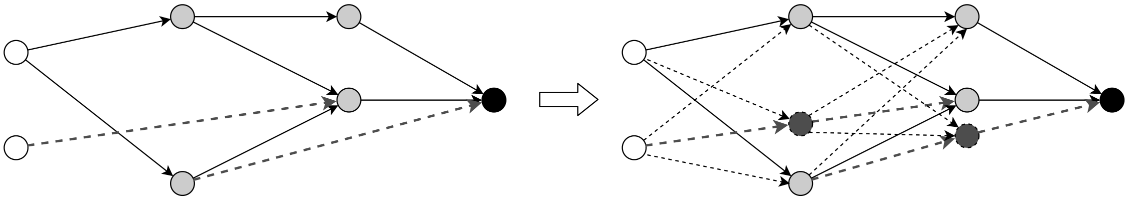

Figure 5: Illustration of how to embed a feedforward network into a multi-layer perceptron, with auxiliary hidden nodes (shown in dark grey).

Proof: The idea is illustrated in Figure 5. For the edges in the directed graph of $f \in \mathcal{F}_{\mathrm{DNN}}$ that connect nodes not in adjacent layers (shown in yellow in Figure 5), one can insert auxiliary hidden units in order to simply "pass forward" the information. The number of such auxiliary "passforward units" is at most the number of offending edges times the depth L (i.e. for each edge, at most L auxiliary nodes are required), and this is bounded by WL. Therefore the width of the MLP network that subsumes the original is upper bounded by $WL + U$ while still maintaining the required embedding that for any $f_\theta \in \mathcal{F}_{\mathrm{DNN}}$, there is a $g_{\theta'} \in \mathcal{F}_{\mathrm{MLP}}$ such that $g_{\theta'} = f_\theta$. In order to match modern practice we only need to show that auxiliary units can be implemented with ReLU activation. This can be done by setting the constant ("bias") term b of each auxiliary unit large enough to ensure $\sigma(\tilde{\boldsymbol{x}}'\boldsymbol{w} + b) = \tilde{\boldsymbol{x}}'\boldsymbol{w} + b$ when $\tilde{\boldsymbol{x}}$ is the input covariates, and then subtracting the same b in the last receiving unit along the path. *Q.E.D.*

Next, we give two properties of the Rademacher complexity (see Mendelson, 2003).

LEMMA 2 (Contraction) *Let $\phi : \mathbb{R} \to \mathbb{R}$ be a Lipschitz function $|\phi(f_1) - \phi(f_2)| \leq L|f_1 - f_2|$, then*

$$\mathbb{E}_\eta R_n\{\phi \circ f : f \in \mathcal{F}\} \leq 2L\mathbb{E}_\eta R_n \mathcal{F}.$$

LEMMA 3 (Dudley's Chaining) *Let $\mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_n)$ denote the metric entropy for class $\mathcal{F}$ (with covering radius δ and metric $\|\cdot\|_n$), then*

$$\mathbb{E}_\eta R_n\{f : f \in \mathcal{F}, \|f\|_n \leq r\} \leq \inf_{0<\alpha<r} \left\{4\alpha + \frac{12}{\sqrt{n}} \int_\alpha^r \sqrt{\log \mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_n)} d\delta \right\} .$$

Furthermore, because $\|f\|_n \leq \max_i |f(\boldsymbol{x}_i)|$, and therefore $\mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_n) \leq \mathcal{N}(\delta, \mathcal{F}|_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n}, \infty)$ and so the upper bound in the conclusions also holds with $\mathcal{N}(\delta, \mathcal{F}|_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n}, \infty)$, where $\mathcal{F}|_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n}$ is the class $\mathcal{F}$ projected onto the data.

The next two results, Theorems 12.2 and 14.1 in Anthony and Bartlett (1999), show that the metric entropy may be bounded in terms of the pseudo-dimension and that the latter is bounded by the Vapnik-Chervonenkis (VC) dimension.

LEMMA 4 *Assume for all $f \in \mathcal{F}$, $\|f\|_\infty \leq M$. Denote the pseudo-dimension of $\mathcal{F}$ as $\mathrm{Pdim}(\mathcal{F})$, then for $n \geq \mathrm{Pdim}(\mathcal{F})$, we have for any δ,*

$$\mathcal{N}(\delta, \mathcal{F}|_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n}, \infty) \leq \left(\frac{2eM \cdot n}{\delta \cdot \mathrm{Pdim}(\mathcal{F})}\right)^{\mathrm{Pdim}(\mathcal{F})} .$$

The following symmetrization lemma bounds the empirical processes term using Rademacher complexity, and is thus a crucial piece of our localization. This is a standard result based on Talagrand's concentration, but here special care is taken with the dependence on the variance.

LEMMA 5 (Symmetrization, Theorem 2.1 in Bartlett et al. (2005)) *For any $g \in \mathcal{G}$, assume that $|g| \leq G$ and $\mathbb{V}[g] \leq V$. Then for every $\gamma > 0$, with probability at least $1 - e^{-\gamma}$*

$$\sup_{g \in \mathcal{G}} \{\mathbb{E}g - \mathbb{E}_n g\} \leq 3\mathbb{E}R_n \mathcal{G} + \sqrt{\frac{2V\gamma}{n}} + \frac{4G}{3}\frac{\gamma}{n} ,$$

and with probability at least $1 - 2e^{-t}$

$$\sup_{g \in \mathcal{G}} \{\mathbb{E}g - \mathbb{E}_n g\} \leq 6\mathbb{E}_\eta R_n \mathcal{G} + \sqrt{\frac{2V\gamma}{n}} + \frac{23G}{3}\frac{\gamma}{n} .$$

The same result holds for $\sup_{g \in \mathcal{G}} \{\mathbb{E}_n g - \mathbb{E}g\}$.

When bounding the complexity of $\mathcal{F}_{\mathrm{DNN}}$, we use the following result. Bartlett et al. (2017) also verify these bounds for the VC-dimension.

LEMMA 6 (Theorem 6 in Bartlett et al. (2017), ReLU case) *Consider a ReLU network architecture $\mathcal{F} =$*

$\mathcal{F}_{\text{DNN}}(W, L, U)$, then the pseudo-dimension is sandwiched as

$$c \cdot WL \log(W/L) \leq \text{Pdim}(\mathcal{F}) \leq C \cdot WL \log W,$$

with some universal constants $c, C > 0$.

For the MLP, we use the following approximation result, Yarotsky (2017) Theorem 1.

Lemma 7 *There exists a network class $\mathcal{F}_{\text{DNN}}$, with ReLU activation, such that for any $\epsilon > 0$:*

(a) *$\mathcal{F}_{\text{DNN}}$ approximates the $W^{\beta,\infty}([-1,1]^d)$ in the sense for any $f_* \in W^{\beta,\infty}([-1,1]^d)$, there exists a $f_n(\epsilon) := f_n \in \mathcal{F}_{\text{DNN}}$ such that*

$$\|f_n - f_*\|_\infty \leq \epsilon,$$

(b) *and $\mathcal{F}_{\text{DNN}}$ has $L(\epsilon) \leq C \cdot (\log(1/\epsilon) + 1)$ and $W(\epsilon), U(\epsilon) \leq C \cdot \epsilon^{-\frac{d}{\beta}}(\log(1/\epsilon) + 1)$.*

Here C only depends on d and β.

For completeness, we verify the requirements on the loss functions, Equation (2.1), for several examples. We first treat least squares and logistic losses, in slightly more detail, as these are used in our subsequent inference results and empirical application.

Lemma 8 *Both the least squares (2.2) and logistic (2.3) loss functions obey the requirements of Equation (2.1). For least squares, $c_1 = c_2 = 1/2$ and $C_\ell = M$. For logistic regression, $c_1 = (2(\exp(M) + \exp(-M) + 2))^{-1}$, $c_2 = 1/8$ and $C_\ell = 1$.*

Proof: The Lipschitz conditions are trivial. For least squares, using iterated expectations

$$\begin{aligned} 2\mathbb{E}\ell(f, \boldsymbol{Z}) - 2\mathbb{E}\ell(f_*, \boldsymbol{Z}) &= \mathbb{E}\left[-2Yf + f^2 + 2Yf_* - f_*^2\right] \\ &= \mathbb{E}\left[-2f_* f(\boldsymbol{x}) + f^2 + 2(f_*)^2 - f_*^2\right] \\ &= \mathbb{E}\left[(f - f_*)^2\right]. \end{aligned}$$

For logistic regression,

$$\mathbb{E}[\ell(f, \boldsymbol{Z})] - \mathbb{E}[\ell(f_*, \boldsymbol{Z})] = \mathbb{E}\left[-\frac{\exp(f_*)}{1 + \exp(f_*)}(f - f_*) + \log\left(\frac{1 + \exp(f)}{1 + \exp(f_*)}\right)\right].$$

Define $h_a(b) = -\frac{\exp(a)}{1+\exp(a)}(b - a) + \log\left(\frac{1+\exp(b)}{1+\exp(a)}\right)$, then

$$h_a(b) = h_a(a) + h_a'(a)(b - a) + \frac{1}{2} h_a''\left(\xi a + (1 - \xi) b\right)(b - a)^2$$

and $h_a''(b) = \frac{1}{\exp(b) + \exp(-b) + 2} \leq \frac{1}{4}$. The lower bound holds as $|\xi f_* + (1 - \xi) f| \leq M$. *Q.E.D.*

Beyond least squares and logistic regression, we give three further examples, discussed in the general language of generalized linear models. Note that in the final example we move beyond a simple scalar outcome.

LEMMA 9 *For a convex function $g(\cdot):\mathbb{R}\to\mathbb{R}$, consider the generalized linear loss function $\ell(f,\boldsymbol{z}) = -\langle y, f(\boldsymbol{x})\rangle + g(f(\boldsymbol{x}))$. The curvature and the Lipschitz conditions in (2.1) will hold given specific $g(\cdot)$. In each case, the loss function corresponds to the negative log likelihood function.*

(a) *Poisson: $g(t)=\exp(t)$, with $f_*(\boldsymbol{x}) = \log\mathbb{E}[y|\boldsymbol{X}=\boldsymbol{x}]$.*

(b) *Gamma: $g(t) = -\log t$, with $f_*(\boldsymbol{x}) = -(\mathbb{E}[y|\boldsymbol{X}=\boldsymbol{x}])^{-1}$.*

(c) *Multinomial Logistic, $K+1$ classes: $g(t)=\log(1+\sum_{k\in K}\exp(t^{[k]}))$, with*

$$\exp(f_*^{[k]}(\boldsymbol{x}))/(1+\sum_{k'\in K}\exp(f_*^{[k']}(\boldsymbol{x}))) = \mathbb{E}[y^{[k]}|\boldsymbol{X}=\boldsymbol{x}].$$

Here $v^{[k]}$ denotes the k-th coordinate of a vector $\boldsymbol{v}$.

PROOF: Denote ∇g, Hessian$[g]$ to be the gradient and Hessian of the convex function g. By the convexity of g, the optimal f_* satisfies $\mathbb{E}[\partial\ell(f_*,\boldsymbol{Z})/\partial f|\boldsymbol{X}=\boldsymbol{x}]=0$, which implies $\nabla g(f_*) = \mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x}]$. If $2c_0 \preceq \text{Hessian}[g(f)] \preceq 2c_1$ for all f of interest, then the curvature condition in (2.1) holds, because

$$\begin{aligned}\mathbb{E}[\ell(f,\boldsymbol{Z})] - \mathbb{E}[\ell(f_*,\boldsymbol{Z})] &= \mathbb{E}[-\langle\nabla g(f_*), f-f_*\rangle + g(f) - g(f_*)] \\ &= \frac{1}{2}\mathbb{E}\langle f-f_*, \text{Hessian}[g(\tilde{f})] f - f_*\rangle \\ &\geq c_0\mathbb{E}\|f-f_*\|^2,\end{aligned}$$

and the parallel argument for $\leq c_1\mathbb{E}\|f-f_*\|^2$. The Lipschitz condition is equivalent to $\|\nabla g(f)\|\leq C'_\ell$ for all f of interest, with bounded Y.

For our three examples in particular, we have the following.

(a) For Poisson regression: $\|\nabla c(f)\| = |\exp(f)| \leq \exp(M)$ and $\text{Hessian}[c(f)] = \exp(f) \in [\exp(-M),\exp(M)]$.

(b) For Gamma regression, bounding $-Y$ above and below is equivalent to $1/M \leq \|f\| \leq M$ and therefore: $\|\nabla c(f)\| = |1/f| \leq M$ and $\text{Hessian}[c(f)] = 1/f^2 \in [1/M^2, M^2]$.

(c) For multinomial logistic regression, with general K, we know $\|\nabla c(f)\| \leq 1$ and $\text{Hessian}[c(f)] = \text{diag}\{z\} - zz^\top$, where $z^{[k]} := \exp(f^{[k]})\left[1+\sum_{k'}\exp(f^{[k']})\right]^{-1}$. One can verify that the eigenvalues are bounded in the following sense, for bounded f,

$$\frac{1}{(1+K\exp(M))^2} \leq \lambda(\text{Hessian}[c(f)]) \leq \frac{\exp(M)}{1+(K-1)\exp(-M)+\exp(M)}.$$

This completes the proof. *Q.E.D.*

Our last result is to verify condition (c) of Theorem 3. We do so using our localization, which may be of future interest in second-step inference with machine learning methods.

LEMMA 10 *Let the conditions of Theorem 3 hold. Then*

$$\mathbb{E}_n\left[(\widehat{\mu}_t(\boldsymbol{x}_i) - \mu_t(\boldsymbol{x}_i))\left(1 - \frac{\mathbb{1}\{t_i=t\}}{\mathbb{P}[T=t|\boldsymbol{X}=\boldsymbol{x}_i]}\right)\right] = o_P\left(n^{-\frac{\beta}{\beta+d}}\log^8 n + \frac{\log\log n}{n}\right) = o_P\left(n^{-1/2}\right).$$

PROOF: Without loss of generality we can take $\bar{p} < 1/2$. The only estimated function here is $\mu_t(\boldsymbol{x})$, which plays the role of f_* here. For function(als) $L(\cdot)$ of the form

$$L(f) := (f(\boldsymbol{x}_i) - f_*(\boldsymbol{x}_i)) \left(1 - \frac{\mathbb{1}\{t_i = t\}}{\mathbb{P}[T = t | \boldsymbol{X} = \boldsymbol{x}_i]}\right),$$

it is true that

$$\mathbb{E}[L(f)] = \mathbb{E}\left[(f(\boldsymbol{X}) - f_*(\boldsymbol{X})) \left(1 - \frac{\mathbb{E}[\mathbb{1}\{t_i = t\} | \boldsymbol{x}_i]}{\mathbb{P}[T = t | \boldsymbol{X} = \boldsymbol{x}_i]}\right)\right] = 0$$

and

$$\begin{aligned} \mathbb{V}[L(f)] &\leq (1/\bar{p} - 1)^2\, \mathbb{E}\left[(f(\boldsymbol{X}) - f_*(\boldsymbol{X}))^2\right] \leq (1/\bar{p} - 1)^2\, \bar{r}^2 \\ |L(f)| &\leq (1/\bar{p} - 1)\, 2M. \end{aligned}$$

For $\bar{r}$ defined in (A.14),

$$\begin{aligned} 18M\mathbb{E}R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r}\} &\leq \bar{r}^2 \\ \mathbb{E}R_n\{L(f) : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r}\} &\leq 2\,(1/\bar{p} - 1)\, \mathbb{E}R_n\{f - f_* : f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r}\} \end{aligned}$$

where the first line is due to $\bar{r} > r_*$, and second line uses Lemma 2.

Then by the localization analysis and Lemma 5, for all $f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r}$, $L(f)$ obeys

$$\begin{aligned} \mathbb{E}_n[L(f)] = \mathbb{E}_n[L(f)] - \mathbb{E}[L(f)] &\leq 6C\bar{r}^2 + \bar{r}\sqrt{\frac{4\,(1/\bar{p} - 1)^2\, t}{n}} + \frac{8\,(1/\bar{p} - 1)\, 3M}{3}\frac{t}{n} \leq 4C\bar{r}^2 \\ &\leq C \cdot \left\{ n^{-\frac{\beta}{\beta + d}} \log^8 n + \frac{\log\log n}{n} \right\}, \\ \sup_{f \in \mathcal{F}, \|f - f_*\|_{L_2(\boldsymbol{X})} \leq \bar{r}} \mathbb{E}_n[L(f)] &\leq C \cdot \left\{ n^{-\frac{\beta}{\beta + d}} \log^8 n + \frac{\log\log n}{n} \right\}. \end{aligned}$$

With probability at least $1 - \exp(-n^{\frac{d}{\beta + d}} \log^8 n)$, $\widehat{f}_{\mathrm{MLP}}$ lies in this set of functions, and therefore

$$\mathbb{E}_n[L(\widehat{f}_{\mathrm{MLP}})] = \mathbb{E}_n\left[(\widehat{f}_{n,H,L}(x) - f_*(x))\left(1 - \frac{1(T = t)}{P(T = t | \boldsymbol{x} = x)}\right)\right] \leq C \cdot \left\{ n^{-\frac{\beta}{\beta + d}} \log^8 n + \frac{\log\log n}{n} \right\},$$

as claimed. *Q.E.D.*