Global Random Maximization of Feedforward Neural Network

Qinghua Li, Lokman Abbas-Turki

Global Random Maximization of Feedforward Neural Network

Qinghua Li*and Lokman A. Abbas-Turki†

October 25, 2023

Abstract

This paper addresses the problem of maximizing functions expressed as a sum of independent terms on a bounded closed domain in $\mathbb{R}^d$. A common approach is to first get a regressed approximation of these functions using neural networks. In this contribution, we propose estimating the maximum by empirically sampling the neural network approximator's output using independent and uniformly distributed inputs. We consider two metrics for quantifying the discrepancy between the sample maximum and the neural network's real maximum: asymptotic distribution and mean-square error. The convergence rate estimation is influenced by the shape of the neural network's output distribution around its maximum point. In some cases, and under minimal assumptions, the convergence rate becomes dimension-dependent. However, with additional assumptions, the convergence rate is free from the curse of dimensionality. The practical implementation of a canonical example illustrates how embracing an estimation bias can substantially enhance the convergence rate. The latter approach paves the way for new theoretical and algorithmic solutions.
Keywords: Global optimization; neural network; Monte Carlo simulation; sample maximum.
MSC codes: 93-08, 90C26, 68T07, 65C05, 62G30.

1 Introduction

Neural Networks (NNs) are gaining widespread popularity across various scientific communities. Their increasing adoption can be attributed, in part, to their approximation power, which was studied theoretically for example in Cybenko (1989); Hornik (1991) and in Kidger and Lyons (2020). NNs also benefit from several computational advantages, notably attributed to their utilization of state-of-the-art linear algebra operations that experience significant enhancements through GPU (Graphics Processing Unit) acceleration Shi et al. (2016). Furthermore, NNs leverage asynchronous Stochastic Gradient Descent (SGD) and minibatch SGD for optimized training processes Mishchenko et al. (2022). Once trained, NNs have high throughput capabilities during the inference stage, particularly when deployed on GPUs, TPUs (Tensor Processing Units), and some ARM-based solutions Castelló et al. (2022). when confronted with the task of searching for the maximum value of a conditional expectation $g(\cdot)$ that is heavy to compute pointwise for z in a bounded closed domain D_0 in $\mathbb{R}^{d_0}$ ($d_0 = 1, 2, ...$), one solution would be to approximate $g(\cdot)$ by a NN function $G(\cdot)$, then evaluating $G(\cdot)$ during the inference phase, one takes the maximum over all the evaluations of $G(\cdot)$.

The actor-critic methods, as detailed in the seminal book Sutton and Barto (1998), represent the established norm when it comes to the approximation of the conditional expectation as well as its maximum value. These methods are well known in the reinforcement learning community, particularly in the context of stationary stochastic problems. Many versions of these actor-critic methods belong to the subcategory of policy gradient methods, these include the widely used DDPG (Deep Deterministic Policy Gradient) presented in Lillicrap et al. (2015) and PPO (Proximal Policy Optimization) originally proposed in Schulman et al. (2017) with a new parallel implementation suited to GPUs given in github. Besides, there is also the subclass of derivative-free optimization methods that include CMA-ES (Covariance Matrix adaptation-Evolution Strategy) reviewed in Hansen (2006). Multiple extensions of the actor-critic methods were also proposed like the one tested in Bachouch et al. (2022).

*School of Statistics, Shandong Technology and Business University, China (202013904@sdtbu.edu.cn). Qinghua Li acknowledges the Doctoral Research Starting Grant No. BS202139 from her affiliated university. The two authors made equal contributions.

†LPSM (UMR 8001), Sorbonne Université, France (lokmane.abbas_turki@sorbonne-universite.fr). The research of L. A. Abbas-Turki has benefited from the support of the Chair *Capital Markets Tomorrow: Modeling and Computational Issues* under the aegis of the Institut Europlace de Finance, a joint initiative of Laboratoire de Probabilités, Statistique et Modélisation (LPSM), Université Paris Cité and Crédit Agricole CIB.

The actor-critic methods are more complicated when compared to what is studied in this paper. Our procedure is simply based on a randomized evaluation of $G(\zeta_i)$, where $\zeta_1, ..., \zeta_n$ are independent random variables identically distributed on $D_0 \subset \mathbb{R}^{d_0}$. It is then our purpose to study the convergence rate of the sample maximum $Y_{(n)}$ of $G(\zeta_1), ..., G(\zeta_n)$ to the real maximum y^* of $G(\cdot)$ on D_0. Denoting F the cumulative distribution function of $G(\zeta_i)$, it is the tail rate of F that analytically determines the convergence rate. Although the convergence rate can be generally established for any function and not only for neural networks, the assumptions needed are easier to check for the feedforward neural network outputs $\{G(\zeta_i)\}_{i=1}^n$ using the results provided in Abbas-Turki et al. (2023). According to Abbas-Turki et al. (2023), when the inputs $\zeta_1, ..., \zeta_n$ are uniformly distributed on D_0 and the neural network $G(\cdot)$ has a Leaky ReLU activation function, the cumulative distribution function F is continuous, piecewise polynomial, and the degree of each polynomial is at most d_0. Let α be the multiplicity of y^* as a root of the polynomial $1-F(y)$. We shall show that the convergence rate is $O\left(n^{-1/\alpha}\right)$ for the asymptotic distribution and $O\left(n^{-2/\alpha}\right)$ for the mean-square error, as $n \to \infty$. With this choice of input distribution and activation function, the most pessimistic case of exponential or sub-exponential right tail (cf. Embrechts et al. (1997)) is ruled out.

Based on Monte Carlo, our approach is also simpler to implement than what is generally considered in the literature and focuses on global maximization of $G(\cdot)$. The presented method does not require advanced Monte Carlo techniques like Stochastic Tunneling (cf. Hamacher (2006)) or Parallel Tempering (cf. Marinari and Parisi (1992)) involving the Metropolis criterion. Indeed, our approach offers a simplicity that is coupled with rapid random assessments of $G(\cdot)$ during the inference phase. It is however admitted that room for further enhancement exists within our approach, like those leveraging importance sampling techniques. The numerical part of this paper also provides an easy improvement idea that opens the door for further theoretical and algorithmic solutions.

The rest of the paper is organized as the following. Section 2 presents the formulation of the considered problem and clarifies its computational motivations. Section 2 also eliminates the unfavorable situation when the distribution of $\zeta_1, ..., \zeta_n$ is not chosen appropriately. When $\zeta_1, ..., \zeta_n$ are independent and uniformly distributed on D_0, Section 3 provides both the asymptotic and the non-asymptotic study of the rate of convergence of $\max_{i=1,...,n}(G(\zeta_i))$ to y^*. On a canonical example, Section 4 illustrates the rate of convergence obtained theoretically and shows that it is possible to improve it numerically with very few changes on the implementation.

2 Random Maximization of a Neural Network

Before studying the rate of convergence in Section 3, Section 2.1 sets the main motivation of our work and the definition of the neural network G. Section 2.2 details the assumptions needed on the architecture of G as well as on the distribution of the inputs $\zeta_1, ..., \zeta_n$.

2.1 Conditional expectation maximization

Let us consider a simplified formulation of a generic problem considered in stochastic control. Given a couple of random variables $(\mathcal{X}, \zeta)$ and a Borel measurable function $\phi : \mathbb{R}^? \times \mathbb{R}^d \to \mathbb{R}$ such that the expectation $\mathbb{E}[|\phi(\mathcal{X}, \zeta)|] < \infty$, we define the function $g(z) = \mathbb{E}[\phi(\mathcal{X}, z)]$ and we assume it to be bounded on the bounded closed domain $D_0 \subset \mathbb{R}^d$. Except for specific models where this expectation can be computed analytically, generally one needs to draw independent realizations $\mathcal{X}_1, ..., \mathcal{X}_m$ of $\mathcal{X}$ and use the Monte Carlo approximation

$$g(z) \approx \frac{1}{m}\sum_{j=1}^{m} \phi(\mathcal{X}_j, z). \tag{2.1}$$

With $g(\cdot)$ approximated as in (2.1), if we wanted also to simulate the maximum value taken by the function g through randomization we would simulate independent realizations $\zeta_1, ..., \zeta_n$ of $\zeta \in D_0$ then pick the approximation

$$\sup_{z \in D_0} \mathbb{E}[\phi(\mathcal{X}, z)] \approx \max_{i=1,...,n} \left(\frac{1}{m}\sum_{j=1}^{m} \phi(\mathcal{X}_j, \zeta_i) \right). \tag{2.2}$$

Example 2.1 *To illustrate the genericity of the considered problem, we present a one-period control example with time variable $t \in \{0, 1\}$. We denote X the controlled state process, whose value at time $t = 0$ is constant $X_0 = x \in \mathbb{R}$, with $X_1 = X_0 + \mathcal{X} + \zeta(X_0)$. $\mathcal{X}$ is a random variable and ζ is the control function of X_0. The control has to be chosen in order to maximize an expected return $\mathbb{E}[\varphi(X_1, \zeta(X_0))] = \mathbb{E}[\phi(x, \mathcal{X}, \zeta(x))]$ with*

$\phi(x,y,z) = \varphi(x+z+y,z)$. *In this case, following approximation* (2.2)*, the optimal control resulting from randomization belongs to* $\underset{i=1,\dots,n}{\operatorname{argmax}} \left(\frac{1}{m} \sum_{j=1}^{m} \phi(x, \mathcal{X}_j, \zeta_i) \right)$.

This example remains quite simple as the control is applied once at the beginning. In a general control problem, the function ϕ is also indexed with respect to the time variable and its computation involves the implementation of a dynamic programming algorithm (cf. Gobet et al. (2005); Huré et al. (2021))

The computation of (2.2) involves $n \times m$ evaluations of ϕ which is very expensive when compared to state-of-the-art methods. Indeed, we rather prefer the simulation of independent realizations $(\mathcal{X}_1, \zeta_1), ..., (\mathcal{X}_m, \zeta_m)$ of the couple $(\mathcal{X}, \zeta)$ in order to get a regressed representation G of g. Considering neural network approximating functions Φ, G minimizes the mean square error (cf. Goodfellow et al. (2016))

$$G \in \underset{\Phi}{\operatorname{argmin}} \frac{1}{m} \sum_{j=1}^{m} |\phi(\mathcal{X}_j, \zeta_j) - \Phi(\zeta_j)|^2. \tag{2.3}$$

The complexity of the numerical minimization (2.3) that involves stochastic gradient descent is proportional to m and thus much cheaper than the computations needed by (2.2). Given that $G(z)$ is also bounded with respect to values taken by z, one can replace the maximization in (2.2) by

$$\sup_{z \in D_0} G(z) \approx \max_{i=1,\dots,n} G(\zeta_i) \tag{2.4}$$

that can be very efficiently implemented during the inference phase. To summarize, the implementation of (2.3) and then (2.4) can be dominated by a complexity that is proportional to $m+n$, in contrast to (2.2) with a complexity proportional to $m \times n$.

The numerical minimization (2.3) is common to almost all actor-critic methods and it is also the origin of the gain in complexity. Indeed, since G is the approximator of $g(z) \approx \frac{1}{\ell} \sum_{k=1}^{\ell} \phi(\mathcal{X}_k, z)$, one could take G to be in the argmin

$$\underset{\Phi}{\operatorname{argmin}} \frac{1}{m} \sum_{j=1}^{m} \Big| \frac{1}{\ell} \sum_{k=1}^{\ell} \phi(\mathcal{X}_k, \zeta_j) - \Phi(\zeta_j) \Big|^2, \tag{2.5}$$

in which case the complexity is proportional to the $\ell \times m$ evaluations of ϕ using the independent realizations $(\mathcal{X}_1, ..., \mathcal{X}_m)$ and $(\zeta_1, ..., \zeta_\ell)$ of $\mathcal{X}$ and ζ. To reduce the unnecessary complexity of the regression (cf. Broadie et al. (2015)), (2.3) replaces (2.5) because

$$\underset{\Phi}{\operatorname{argmin}}\, \mathbb{E} \Big| \mathbb{E}(\phi(\mathcal{X}, \zeta)|\zeta) - \Phi(\zeta) \Big|^2 = \underset{\Phi}{\operatorname{argmin}}\, \mathbb{E} \Big| \phi(\mathcal{X}, \zeta) - \Phi(\zeta) \Big|^2. \tag{2.6}$$

Remark 2.1 *In a typical control problem (cf. Bertsekas (2012)), the focus is rather on the optimization of functions expressed in terms of the state vector involving G. The admissible controls are generally assumed to be deterministic functions of a state vector. Iterative methods can be implemented, interleaving the estimation of the value function given the current policy with the enhancement of the said policy.*

Although the implementation of (2.3) is preferred to the implementation of (2.5), it is important to check whether the real $g(z) = \mathbb{E}[\phi(\mathcal{X}, z)]$ is close enough to its approximator $G(z)$. Presented in Abbas-Turki et al. (2023), the twin method allows to compute the mean-square distance between G and g. Without loss of generality, assuming the independence of $\mathcal{X}$ and ζ and considering two independent copies $\mathcal{X}^{(1)}$ and $\mathcal{X}^{(2)}$ of $\mathcal{X}$, the twin method uses the following identity

$$\mathbb{E} \Big| \mathbb{E}(\phi(\mathcal{X}, \zeta)|\zeta) - \Phi(\zeta) \Big|^2 = \mathbb{E} \left(\phi(\mathcal{X}^{(1)}, \zeta) \phi(\mathcal{X}^{(2)}, \zeta) - [\phi(\mathcal{X}^{(1)}, \zeta) + \phi(\mathcal{X}^{(2)}, \zeta)] \Phi(\zeta) + \Phi^2(\zeta) \right). \tag{2.7}$$

Consequently, the following estimation of the mean-square error can be computed

$$\mathbb{E} \Big| g(\zeta) - G(\zeta) \Big|^2 \approx \frac{1}{m} \sum_{j=1}^{m} \left(\phi(\mathcal{X}_j^{(1)}, \zeta_j) \phi(\mathcal{X}_j^{(2)}, \zeta_j) - [\phi(\mathcal{X}_j^{(1)}, \zeta_j) + \phi(\mathcal{X}_j^{(2)}, \zeta_j)] G(\zeta_j) + G^2(\zeta_j) \right). \tag{2.8}$$

As long as one can simulate independent triplets $(\zeta_1, \mathcal{X}_1^{(1)}, \mathcal{X}_1^{(2)}), ..., (\zeta_m, \mathcal{X}_m^{(1)}, \mathcal{X}_m^{(2)})$, the approximation (2.8) can be implemented with a complexity proportional to m, which is cheaper than $\ell \times m$ required in (2.5). Moreover, in the general setting, for each $j \in \{1, ..., m\}$, the triplet ζ_j, $\mathcal{X}_j^{(1)}$ and $\mathcal{X}_j^{(2)}$ must not be independent; only $\mathcal{X}_j^{(1)}$ and $\mathcal{X}_j^{(2)}$ need to be independent conditionally to ζ_j.

From what has been presented above, a neural network function G, as an approximation of a conditional expectation function $\mathbb{E}[\phi(\mathcal{X},\zeta)|\zeta=z]$, can be obtained from the minimization (2.3) and tested using (2.8) with complexity only proportional to m. Once G is obtained and its accuracy is satisfactory, one can perform (2.4) using very effective n evaluations during the inference phase.

In contrast to minimization (2.3) largely used in the actor-critic methods, (2.4) is generally implemented only when z takes its values in a finite set. This is however not our case since we assume that z belongs to a bounded closed domain $D_0 \subset \mathbb{R}^{d_0}$ ($d_0 = 1, 2, ...$). In this setting, with the generic simplified formulation presented in the previous subsection, usual actor-critic methods introduce another neural network Ξ for the maximization of the value of $G(\Xi)$. In this paper, we simulate independent, identically distributed realizations $\zeta_1, ..., \zeta_n$ whose sample space is identically the whole domain D_0. These realizations are then used for the evaluation of the output of the neural network G.

The neural network G to be considered is a feedforward neural network $G : \mathbb{R}^{d_0} \to \mathbb{R}$ defined as compositions of affine transformations and an activation function a. Detailed descriptions of such feedforward neural networks are available in Section 2 of Abbas-Turki et al. (2023). Common types of activation functions include

– Leaky ReLU:

$$a(z) = a_+ \max\{z, 0\} + a_- \min\{z, 0\},\ z \in \mathbb{R}, \tag{2.9}$$

where the parameters a_+ and a_- are two positive real numbers;

– Sigmoid:

$$a(z) = \frac{1}{1+e^z},\ z \in \mathbb{R}; \tag{2.10}$$

– Soft plus

$$a(z) = \log(1+e^z),\ z \in \mathbb{R}; \tag{2.11}$$

– Hypertangent

$$a(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}},\ z \in \mathbb{R}. \tag{2.12}$$

Subsection 2.2 will study the asymptotic distribution and estimate the mean-square error of the sample maximum within the class of feedforward neural networks.

2.2 Performance of the randomization method

We propose the randomization method of maximizing a neural network G with uniform inputs $\zeta_1, \zeta_2, \cdots$. The suggested activation function type is Leaky ReLU. This subsection evaluates this proposed method for the performance of its sample maximum, consisting of the asymptotic distribution and the mean-square error. Based on the performance of maximizing one neural network, a possible improvement in the convergence rate is provided. An example at the end of this subsection shows the reason why the proposed method is preferred to directly maximizing the original function g.

Assumption 2.1 *(i) The activation function a is continuous.*
(ii) The activation function a is strictly increasing.
(iii) The activation function a is d_0-th continuously differentiable in $\mathbb{R}$.

Property 2.1 *Under Assumption 2.1 (i), the neural network G is a continuous function, achieving a finite maximum y^* and a finite minimum y_* on D_0.*

All activation functions of the above types (2.9)-(2.12) satisfy Assumption 2.1 (i) and (ii), and all but Leaky ReLU satisfy Assumption 2.1 (iii). By its inductive construction, the neural network G is bounded by Property 2.1.

To simplify notations, in this section we denote $Y_n := G(\zeta_n)$ and the "sample maximum" $Y_{(n)} := \max\{Y_1, \cdots, Y_n\}$ as in (3.1). The so-defined random variables $Y_1, Y_2, \cdots$ satisfy Assumption 3.1, with y^* and y_* from Property 2.1. By Theorem 3.1, $Y_{(n)} = \max_{i=1,\ldots,n}(G(\zeta_i))$ converges almost surely to y^*, as $n \to \infty$.

Characterization of the asymptotic distribution of the sample maximum will rely on the Extreme Value Theory. Ideally, we would like to find a normalizing constant $c_n > 0$, such that the sequence

$$\{c_n^{-1}(Y_{(n)} - y^*)\}_{n=1}^{\infty} \tag{2.13}$$

converges to some non-degenerate limiting distribution H.

Definition 2.1 *(maximum domain of attraction) [Definition 3.3.1 on page 128 of Embrechts et al. (1997)] We say the random variable Y_1 or its distribution F belongs to the maximum domain of attraction of an extreme value distribution H, if there exist real constants c_n and d_n such that the sequence*

$$\{c_n^{-1}(Y_{(n)} - d_n)\}_{n=1}^{\infty} \tag{2.14}$$

converges to H in distribution.

Theorem 2.1 *(Fisher-Tippett Theorem) [Theorem 3.2.3 on page 121 of Embrechts et al. (1997)] The extreme value distribution H in Definition 2.1 belongs to one of the three types: Fréchet, Weibull and Gumbel.*

Considering in addition that the sample maximum $Y_{(n)}$ is bounded from above by the maximum y^*, the limiting distribution of (2.13) is supported within the negative real line. It follows that the only possible limiting distribution is Weibull or Gumbel types, depending on the tail rate of the sample distribution.

The cumulative distribution function Ψ_α of a Weibull distribution with parameter α is

$$\Psi_\alpha(y) = \begin{cases} \exp(-(-y)^\alpha), \, y \leq 0; \\ 1, \, y > 0, \end{cases} \tag{2.15}$$

for some parameter $\alpha > 0$. The cumulative distribution function Λ of a Gumbel distribution is

$$\Lambda(y) = \exp(-\exp(-y)), \, y \in \mathbb{R}. \tag{2.16}$$

The mean-square error between the sample maximum $Y_{(n)}$ and the maximum y^* is $\mathbb{E}\left[R_n^2\right]$, where the random variable R_n is defined as

$$R_n := Y_{(n)} - y^*, \text{ for } n = 1, 2, \cdots, \infty. \tag{2.17}$$

Theorem 2.2 *[Theorem 2.4 in Abbas-Turki et al. (2023)] If G is a Leaky ReLU neural network with its activation function defined in 2.9 and ζ is a uniformly distributed random variable on the polytope domain D_0, then the cumulative distribution function $F(y)$ of $G(\zeta)$ is continuous with respect to y, and is piecewise polynomial. Each piece of these polynomials is of degree at most d_0.*

With LeakyReLU neural network and uniform random input, let $F(y)$ be the cumulative distribution function in Theorem 2.2 and α be the multiplicity of y^* as a root of $1 - F(y)$. Then Assumption 3.3 is true. By Theorem 3.2, expression (2.13) converges to the Weibull distribution with parameter α; the normalizing constant c_n is of the order $O\left(n^{-1/\alpha}\right)$, as $n \to \infty$. The asymptotic confidence interval of the maximum y^* is specified by Corollary 3.1. The mean-square error $\mathbb{E}\left[R_n^2\right]$ has the estimation (3.12) for every positive integer n, and $\mathbb{E}\left[R_n^2\right] \leq O\left(n^{-2/\alpha}\right)$ as $n \to \infty$.

When the neural networks G has piecewise affine activation functions and the input random variable ζ has a histogram distribution, the conclusion in Theorem 2.2 is still true by Remark 2.5 in Abbas-Turki et al. (2023). Hence the limiting behaviors in the above paragraph remain.

Example 2.2 takes a general shallow neural network to show that the value α in the limiting behavior estimations can indeed take any value in $\{1, \cdots, d_0\}$.

Example 2.2 *Let $D_0 = [0,1]^{d_0}$. The integer d takes values in $\{1, 2, \cdots, d_0\}$. The neural network has the expression*

$$G(z) = a\left(\sum_{k=1}^{d} z_k\right), \text{ for } z = (z_1, \cdots, z_{d_0}) \in D_0. \tag{2.18}$$

The maximum of this neural network is $y^* = a(d)$, achieved by $z^* = (z_1^*, \cdots, z_{d_0}^*)$, where $z_1^* = \cdots z_d^* = 1$ and $z_{d+1}^*, \cdots, z_{d_0}^*$ taking any value in $[0,1]$. We suppose that the activation function a is either Leaky ReLU or satisfies Assumption 2.1.

(i) *The sequence* (2.13) *converges to a Weibull distribution with parameter $\alpha = d$. The normalizing constant $c_n = a'(d)(d!)^{1/d} n^{-1/d}$.*

(ii) *There exist positive constants C_1 and C_2, such that*

$$\mathbb{E}\left[R_n^2\right] \leq C_1(1 - 1/d!)^n + C_2 n Beta(1 + 2/d, n) \tag{2.19}$$

for every positive integer n, where the Beta function is defined as

$$Beta(p, q) := \int_0^1 x^{p-1}(1-x)^{q-1} dx \tag{2.20}$$

for any positive numbers p and q, and $\mathbb{E}\left[R_n^2\right] \leq O\left(n^{-2/d}\right)$ as $n \to \infty$.

The estimation of the mean-square error in 2.19 is an upper bound. We shall illustrate the real convergence rate in a simplest case. In Example 2.2, we take the Leaky ReLU activation function (2.9), with $a_+ = 1$ and any value of a_-. Let $\{\zeta_n = (\zeta_{n1}, \cdots, \zeta_{nd_0})\}_{n=1}^{\infty}$ be independent uniform random vectors on $[0,1]^{d_0}$. For every d in $\{1, 2, \cdots, d_0\}$ and $n = 1, 2, \cdots$, we define each sample

$$Y_n^{(d)} := G(\zeta_n) = a\left(\sum_{k=1}^{d} \zeta_{nk}\right) = \sum_{k=1}^{d} \zeta_{nk} \tag{2.21}$$

and the sample maximum

$$Y_{(n)}^{(d)} := \max\{Y_1^{(d)}, \cdots, Y_n^{(d)}\}. \tag{2.22}$$

By Theorem 3.1, $Y_{(n)}^{(d)}$ converges to d and $\frac{1}{d}Y_{(n)}^{(d)}$ converges to one almost surely, as $n \to \infty$. Figure 1 shows the convergence of $\frac{1}{d}Y_{(n)}^{(d)}$ to one, for $d_0 \geq 7$ and $d = 1, 2, 3, 4, 7$. Overall, the convergence is slower for lager d.

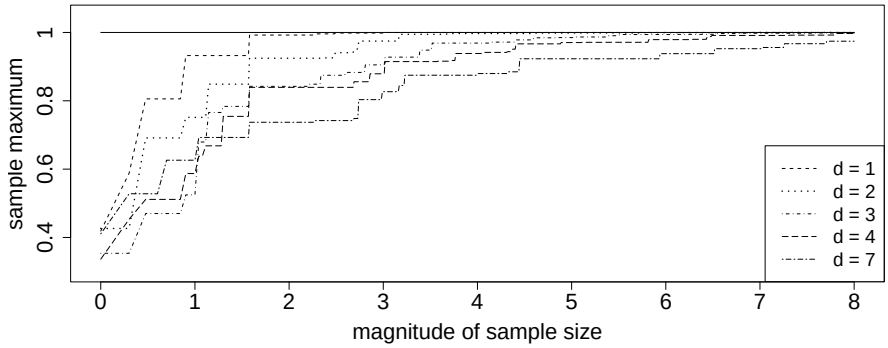

Figure 1: A simulated path of $\frac{1}{d}Y_{(n)}^{(d)}$ versus magnitude $\log_{10}^n$.

This simplest case suggests that, not only the estimated but also the real convergence rate can differ for different choices of the neural network G. Intuitively, the convergence rate depends on the shape of a neural network near its maximum. If the pre-image set on which the neural network takes values near y^* is smaller, then the converge is slower. In practice, there are multiple fitted neural networks such that the loss function is small enough. One way to possibly speed up the convergence is to simultaneously maximize the multiple fitted neural networks and report the maximum of the maxima. This improvement will be introduced in detail in Subsection 4.2.

As presented in Subsection 2.1, replacing an arbitrary function g with a neural network G has the benefit of decreasing the complexity of numerical computations. In this subsection, the benefit of using G is expressed in terms of the interpretation of the assumptions needed to establish the convergence results of Section 3. Indeed, with G, the assumptions can be translated in terms of the specification of both the neural network architecture and the distribution of the sequence $\zeta_1, ..., \zeta_n$. However, without a priori knowledge about the function g approximated by (2.1), it is impossible to check the requested assumptions. In particular, we might find ourselves in very pessimistic situations like Example 2.3.

Example 2.3 *We consider the function $g(z) = \ln(z)/(\ln(z) - 1)$ for $z \in (0, 1]$, and extend it to $g(0) = 1$ by continuity. The maximum of g on $D_0 = [0, 1]$ is $y^* = 1$. Let the independent realizations $\zeta_1, ..., \zeta_n$ be uniformly distributed on $[0, 1]$. Let Y_n denote $g(\zeta_n)$ and $Y_{(n)}$ denote the sample maximum as in (3.1), for $n = 1, 2, \cdots$.*

(i) *With $c_n = (1 + \ln(n))^{-2}$ and $d_n = \dfrac{\ln(n)}{1 + \ln(n)}$, Y_1 belongs to the domain of attraction of the Gumbel distribution as in Definition 2.1.*

(ii) *With R_n defined as in (2.17), for any $\epsilon \in (0, 1)$, there exists a positive constant $C(\epsilon)$, such that the mean-square error $\mathbb{E}\left[R_n^2\right] > \dfrac{n}{C(\epsilon)} Beta(1 - \epsilon, n)$, where the Beta function is defined in (2.20).*

The worst cases of maximizing an arbitrary original function g is outperformed by our proposed choice of maximizing an approximating Leaky ReLU neural neural network G with uniform inputs. For the function

g as in Example 2.3, the convergence of its sample maximum is strikingly slow. This example is in dimension one. However, its convergence is slower than that of our proposed choice in any dimension d_0. According to Example 2.3 (i), Theorem 2.2 and Theorem 3.2, the normalizing constant c_n in the two cases is respectively $(1+\ln(n))^{-2}$ and $O\left(n^{-1/\alpha}\right)$, for some positive integer $\alpha \leq d_0$. In the latter case, Theorem 3.5 gives an estimate of the mean-square error. On the right hand side of the inequality (3.12), the rate $nBeta(1+2/\alpha, n)$ in the second summand dominates. Figure 2 compares this rate with the lower-bound rate $nBeta(1-\epsilon, n)$ in Example 2.3 (ii). The estimated mean-square error is obviously faster in the latter case.

Besides, the limit in distribution in Example 2.3 (i) is biased. If considering the convergence to the real max $y^* = 1$, seeing from the identity $c_n^{-1}(Y_{(n)} - 1) = c_n^{-1}(Y_{(n)} - d_n) - (1 + \ln(n))$, for large n the distribution of $c_n^{-1}(Y_{(n)} - 1)$ is roughly a Gumbel random variable centered at $-(1+\ln(n))$.

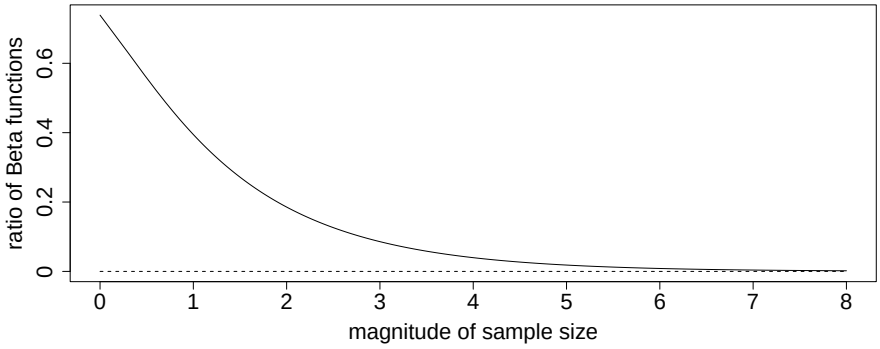

Figure 2: The ratio $Beta\left(1+\frac{2}{\alpha}, n\right)/Beta(1-\epsilon, n)$, with $\alpha = 7$ and $\epsilon = 0.05$, versus sample size magnitude $\log_{10}^n$.

3 Limiting theorems of the sample maximum

Assumption 3.1 *The random variables $Y_1, Y_2, \cdots$ are independent and identically distributed, each having finite maximum y^* and finite minimum y_*.*

Let F denote the cumulative distribution function of these random variables and f denote their probability density function when it exists. By the "maximum" ("minimum") of a random variable in Assumption 3.1, we mean the supremum (infimum) of the values at which its cumulative distribution function is smaller than one (greater than zero). The "sample maximum"

$$Y_{(n)} := \max\{Y_1, \cdots, Y_n\} \tag{3.1}$$

is the largest order statistic among $Y_1, \cdots, Y_n$, for $n = 1, \cdots, \infty$. Under Assumption 3.1, this section will discuss the limiting behaviors of the sample maximum $Y_{(n)}$ as $n \to \infty$.

Theorem 3.1 *[pages 114-115 of Embrechts et al. (1997)] The sample maximum $Y_{(n)}$ converges to the maximum y^* almost surely, as $n \to \infty$.*

3.1 Asymptotic distribution

The limiting distribution is determined by the tail behaviour of the distribution of Y_1 near it maximum y^*.

Assumption 3.2 (i) *The density function f exists and, on the interval $(y^* - \epsilon, y^*)$ for some positive number ϵ, is positive.*

(ii) *(von Mises condition) The limit $\lim_{y \to y^*-} (y^* - y)f(y)/(1 - F(y))$ exists and is positive.*

Property 3.1 (i) *[Theorem 3.3.12 on page 135 of Embrechts et al. (1997)] If F is in the maximum domain of attraction of Ψ_α, then the normalizing constants in Definition 2.1 can be chosen as $d_n = y^*$ and c_n such that*

$$1 - F(y^* - c_n) = \frac{1}{n}. \tag{3.2}$$

Besides, [page 154 of Embrechts et al. (1997)] the normalizing constant c_n and the parameter α are related by $c_n = n^{-1/\alpha}S(n)$, where S is a slowly varying function satisfying $\lim_{n\to\infty} S(tn)/S(n) = 1$, for any positive real number t.

(ii) *[Corollary 3.3.13 on page 136 of Embrechts et al. (1997)] Under Assumption 3.2, F is in the maximum domain of attraction of Ψ_α, with*

$$\alpha = \lim_{y\to y^*-} \frac{(y^* - y)f(y)}{1 - F(y)}. \tag{3.3}$$

We shall focus on the case of polynomial distributions.

Assumption 3.3 (i) *The cumulative distribution function F satisfies $1-F(y) = O\left((y^* - y)^\alpha\right)$ as $y \to y^*-$, with some positive number α.*

(ii) *The cumulative distribution function F is a polynomial of degree d on the interval $[y^*-\epsilon, y^*]$, for some positive integer d and some positive number ϵ. The polynomial $1 - F(y)$ has y^* as its root with multiplicity $\alpha \in \{1, \cdots, d\}$.*

Theorem 3.2 *Under Assumption 3.3* (i) *or* (ii)*, the sequence* (2.13) *converges to a Weibull distribution with parameter α; the normalizing constant c_n is of the order $O\left(n^{-1/\alpha}\right)$, as $n \to \infty$.*

Given the limiting Weibull distribution, we may calculate an asymptotic confidence interval for the maximum y^* in terms of the sample maximum $Y_{(n)}$.

Corollary 3.1 *Suppose that the sequence* (2.13) *converges in distribution to Weibull with parameter $\alpha > 0$. Let λ be a fraction valued in $(0, 1)$. Then as $n \to \infty$, an asymptotic λ-confidence interval for y^* is*

$$\left[Y_{(n)}, Y_{(n)} - c_n\Psi_\alpha^{-1}(1 - \lambda)\right]. \tag{3.4}$$

3.2 Mean-square error

This subsection studies the mean-square error $\mathbb{E}\left[R_n^2\right]$, with $R_n = Y_{(n)} - y^*$ defined in (2.17) and y^* the maximum as in Assumptioin 3.1.

A pioneer paper that discusses the distance between a quantile of a distribution and the sample quantile is Bahadur (1966). Under certain conditions on the cumulative distribution function F of the samples, Duttweiler (1973) continues to estimate the mean-square error as no larger than $O\left(n^{-\frac{3}{2}}\right)$. The idea of the estimation starts with representing the sample as a function Q of a Uniform[0,1] random variable. Then the distance is estimated respectively for the major possible observations and for the tail of the distribution.

We apply the same idea as in Duttweiler (1973) to give two estimates of the mean-square error for the maximum. One estimate generalizes the conditions and the other has a sharper convergence rate. The estimation includes a non-asymptotic result that applies to any (large enough) sample size n.

The function Q that maps the Uniform[0,1] random variable to the samples is the quantile function

$$Q : [0, 1] \to [y_*, y^*], u \mapsto Q(u), \tag{3.5}$$

of the distribution F, defined as

$$Q(u) := \inf\left\{y \in [y_*, y^*] \,|F(y) \geq u\right\}, \; u \in [0, 1]. \tag{3.6}$$

Theorem 3.3 *The mean-square error $\mathbb{E}\left[R_n^2\right]$ has the following estimates:*

(i) *for any real number $\delta \in (0, 1)$ and for every positive integer n, it holds that*

$$\mathbb{E}\left[R_n^2\right] \leq (y^* - y_*)^2 ne^{-n^{1-\delta}} + \left(y^* - Q\left(1 - n^{-\delta}\right)\right)^2; \tag{3.7}$$

(ii) *as $n \to \infty$, it holds that*

$$\mathbb{E}\left[R_n^2\right] \leq O\left(ne^{-n^{1-\delta}}\right) + O\left(\left(y^* - Q\left(1 - n^{-\delta}\right)\right)^2\right). \tag{3.8}$$

In the inequalities (3.7) and (3.8) of Theorem 3.3, the quantity $y^* - Q\left(1 - n^{-\delta}\right)$ is the length of the right tail with probability $n^{-\delta}$ for the distribution F. The thinner the right tail, the larger this quantity. A thinner right tail is equivalent to a smaller probability to get a sample near the maximum y^*, thus implying a larger distance between the sample value and y^*.

Proposition 3.1 *If the cumulative distribution function F is continuous in a neighbourhood of y^* and satisfies Assumption 3.3* (i)*, then as $n \to \infty$, the convergence rate in inequality* (3.8) *becomes*

$$\mathbb{E}\left[R_n^2\right] \leq O\left(n^{-\frac{2\delta}{\alpha}}\right). \tag{3.9}$$

If allowing more assumptions on the tail behavior of the sample distribution, the estimates in Theorem 3.3 can be improved.

Assumption 3.4 *(i) The density function f of the random variable Y_1 exists in a neighborhood $(y^*-\epsilon_F, y^*]$, for some $\epsilon_F > 0$, and is continuously differentiable in this neighborhood.*
(ii) The value $f(y^*)$ is non-zero.

Theorem 3.4 *Under Assumption 3.4, the mean-square error $\mathbb{E}\left[R_n^2\right]$ has the following estimates:*
(i) *there exists an $\epsilon_0 \in (0, \epsilon_F]$, such that for any positive real number $\delta \in (0,1)$ and for every positive integer $n > (1-F(y^*-\epsilon_0))^{-\delta}$, it holds that*

$$\begin{aligned}\mathbb{E}\left[R_n^2\right] \leq & \frac{4}{f^2(y^*)} \cdot \frac{1}{(n+1)(n+2)} \\ & + \left((y^*-y_*)^2 + \frac{1}{f^2(y^*)}\right) n e^{-n^{1-\delta}} \\ & + \frac{1}{4}\left(\sup_{y\in(Q(1-n^{-\delta}),y^*]}\left\{\frac{(f'(y))^2}{f^6(y)}\right\}\right) n^{-4\delta};\end{aligned} \tag{3.10}$$

(ii)*as $n \to \infty$, it holds that*

$$\mathbb{E}\left[R_n^2\right] \leq O(n^{-2}). \tag{3.11}$$

Assumption 3.4 consists of the very assumptions taken in Duttweiler (1973). Compared to the $O\left(n^{-\frac{3}{2}}\right)$ convergence rate in that paper, Theorem 3.4 provides a faster convergence rate for the mean-square error in the case of approximating the maximum.

Theorem 3.5 *Under Assumption 3.3* (ii)*, there exists a positive constant $C(\epsilon, \alpha)$, such that*

$$\mathbb{E}\left[R_n^2\right] \leq (y^*-y_*)^2(F(y^*-\epsilon))^n + C(\epsilon,\alpha) n Beta(1+2/\alpha, n), \tag{3.12}$$

for every positive integer n, where the Beta function is defined in (2.20)*, and $\mathbb{E}\left[R_n^2\right] \leq O\left(n^{-2/\alpha}\right)$ as $n \to \infty$.*

4 Numerical Illustrations

This section illustrates the convergence rate in section 3 with a specific type of examples, and suggest an improvement of the convergence rate. For some non-negative integer r, the random variables $Y_1, Y_2, \cdots$ at the beginning of section 3 are assumed to have the probability density function

$$f_r(y) := \begin{cases} \dfrac{r+1}{(y^*-y_*)^{r+1}}(y^*-y)^r, \ y_* < y < y^*; \\ 0, \text{ elsewhere,} \end{cases} \tag{4.1}$$

plotted in Figure 3. Unless specified otherwise, the illustrations take $y^* = 3$ and $y_* = 1$.

The corresponding cumulative distribution function is

$$F_r(y) := \begin{cases} 0, \ y \leq y_*; \\ 1 - \left(\dfrac{y^*-y}{y^*-y_*}\right)^{r+1}, \ y_* < y < y^*; \\ 1, \ y \geq y^*. \end{cases} \tag{4.2}$$

This density f_r satisfies Assumptions 3.1, 3.2 and 3.3, therefore all the results in Section 3 but Theorem 3.4 applies. Especially when $r = 0$, Assumption 3.4 is satisfied and thus Theorem 3.4 holds.

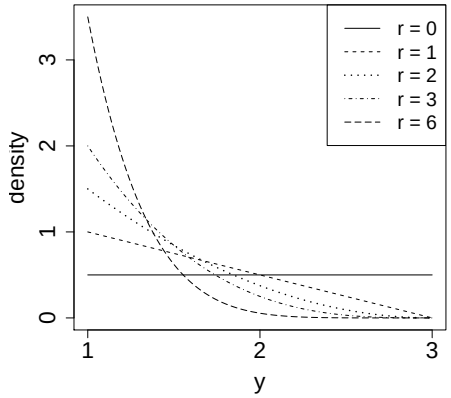

Figure 3: Density f_r for $r = 0, 1, 2, 3, 6$.

r	magnitude	c_n	95% ACI	99% ACI
	2	2.00e-02	9.21e-02	5.99e-02
0	5	2.00e-05	9.21e-05	5.99e-05
	8	2.00e-08	9.21e-08	5.99e-08
	3	6.32e-02	1.36e-01	1.09e-01
1	9	6.32e-05	1.36e-04	1.09e-04
	15	6.32e-08	1.36e-07	1.09e-07
	4	9.28e-02	1.54e-01	1.34e-01
2	13	9.28e-05	1.54e-04	1.34e-04
	22	9.28e-08	1.54e-07	1.34e-07
	5	1.12e-01	1.65e-01	1.48e-01
3	17	1.12e-04	1.65e-04	1.48e-04
	29	1.12e-07	1.65e-07	1.48e-07
	9	1.04e-01	1.29e-01	1.21e-01
6	30	1.04e-04	1.29e-04	1.21e-04
	51	1.04e-07	1.29e-07	1.21e-07

Table 1: Asymptotic convergence rates.

4.1 Quantification of convergence rates

Property 3.1 (i) and Theorem 3.2 (i) suggest that the sequence (2.13) has the Weibull distribution Ψ_α as its limiting distribution, with $c_n = (y^* - y_*)n^{-\frac{1}{r+1}}$ and $\alpha = r + 1$. Table 1 shows the normalizing constant c_n and the lengths of asymptotic confidence intervals (ACI) for different values of the root multipilicity r and the sample size n (counted by its magnitude $\log_{10} n$, same below).

According to calculation by hand, the mean-square error $\mathbb{E}[R_n^2]$ with R_n defined in (2.17) has the exact expression

$$\mathbb{E}[R_n^2] = (y^* - y_*)^2 n Beta\left(1 + \frac{2}{r+1}, n\right), \tag{4.3}$$

where the Beta function is defined in (2.20).

The quantile function in (3.6) becomes

$$Q_r(u) = y^* - (y^* - y_*)(1-u)^{\frac{1}{r+1}},\ u \in [0,1]. \tag{4.4}$$

Theorem 3.3 (i) gives the estimation

$$\mathbb{E}\left[R_n^2\right] \le (y^* - y_*)^2 \left(n e^{-n^{1-\delta}} + n^{-\frac{2\delta}{r+1}}\right), \tag{4.5}$$

for any positive integer n. Especially, when $r = 0$, Theorem 3.4 (i) gives the estimation

$$\mathbb{E}\left[R_n^2\right] \leq 2\left(y^* - y_*\right)^2 \left(\frac{2}{(n+1)(n+2)} + ne^{-n^{1-\delta}}\right). \tag{4.6}$$

From the first sentence in the proof of Theorem 3.4 given in Section 6.2, the number ϵ_0 in this theorem can be chosen as $\epsilon_0 = y^* - y_*$. Hence the estimation (4.6) is true for any integer $n > 1$. Table 2 shows the mean-square errors, where "real", "estimated" and "non-vanishing density" indicate respectively the values on the right hand side of (4.3), (4.5) and (4.6). Now that the estimations (4.5) and (4.6) are valid for any $\delta \in (0, 1)$, the values in the "estimated" and "non-vanishing density" columns are respectively the infima of these two estimations over $\delta \in \{0.01, 0.02, \cdots, 0.99\}$.

r	magnitude	real	estimated	non-vanishing density
0	1	6.06e-02	1.25e+00	1.26e-01
	2	7.77e-04	5.50e-02	1.55e-03
	4	8.00e-08	2.55e-05	1.60e-07
1	1	3.64e-01	2.24e+00	NA
	4	4.00e-04	7.83e-03	
	7	4.00e-07	1.39e-05	
2	2	1.67e-01	8.52e-01	NA
	6	3.61e-04	3.48e-03	
	11	1.68e-07	2.35e-06	
3	3	1.12e-01	4.63e-01	NA
	9	1.12e-04	7.44e-04	
	15	1.12e-07	1.00e-06	
6	5	1.34e-01	3.47e-01	NA
	15	1.86e-04	6.77e-04	
	26	1.34e-07	5.86e-07	

Table 2: Mean-square errors.

In addition to the results in Section 3, the number of simulations needed to reach a certain accuracy can be estimated from a geometric distribution. For any number $\epsilon \in (0, y^* - y_*)$, the random variable

$$N_{r,\epsilon} := \inf\{n \in \mathbb{N} \,|\, Y_{(n)} > y^* - \epsilon\} \tag{4.7}$$

is the minimal sample size n needed such that the sample maximum $Y_{(n)}$ is less than ϵ away from the maximum y^*, which has a geometric distribution with success probability

$$p_{r,\epsilon} := \mathbb{P}(Y > y^* - \epsilon) = 1 - F_r(y^* - \epsilon) = \left(\frac{\epsilon}{y^* - y_*}\right)^{r+1}. \tag{4.8}$$

Tables 3, 4 and 5 show the success probability, expectation and quantiles of this geometric distribution for different accuracies $\epsilon = 0.1, 10^{-4}, 10^{-7}$.

r	$p_{r,\epsilon}$	$\mathbb{E}[N_{r,\epsilon}]$	2.5% quantile	97.5% quantile
0	5.00e-02	2.00e+01	0.00e+00	7.10e+01
1	2.50e-03	4.00e+02	1.00e+01	1.47e+03
2	1.25e-04	8.00e+03	2.02e+02	2.95e+04
3	6.25e-06	1.60e+05	4.05e+03	5.90e+05
6	7.81e-10	1.28e+09	3.24e+07	4.72e+09

Table 3: Sample size needed for $\epsilon = 0.1$.

r	$p_{r,\epsilon}$	$\mathbb{E}[N_{r,\epsilon}]$	2.5% quantile	97.5% quantile
0	5.00e-05	2.00e+04	5.06e+02	7.38e+04
1	2.50e-09	4.00e+08	1.01e+07	1.48e+09
2	1.25e-13	8.00e+12	2.03e+11	2.95e+13
3	6.25e-18	1.60e+17	4.05e+15	5.90e+17
6	7.81e-31	1.28e+30	3.24e+28	4.72e+30

Table 4: Sample size needed for $\epsilon = 1e-4$.

r	$p_{r,\epsilon}$	$\mathbb{E}[N_{r,\epsilon}]$	2.5% quantile	97.5% quantile
0	5.00e-08	2.00e+07	5.06e+05	7.38e+07
1	2.50e-15	4.00e+14	1.01e+13	1.48e+15
2	1.25e-22	8.00e+21	2.03e+20	2.95e+22
3	6.25e-30	1.60e+29	4.05e+27	5.90e+29
6	7.81e-52	1.28e+51	3.24e+49	4.72e+51

Table 5: Sample size needed for $\epsilon = 1e-7$.

4.2 Simulation and improvement

In Section 3, both the asymptotic distribution and the mean-square error are determined by the tail behavior of the distribution F. Specifically, when the tail is polynomial, the determining factor is the multiplicity of the maximum y^* as a root of the polynomial. These results inspire a possible improvement of the convergence rates by perturbing the distribution F, likely at the price of an acceptable bias.

In the case of approximating a function g with a neural network G as in Section 2.1, there exist several candidate neural networks $G^{(1)}, \cdots, G^{(M)}$ such that the quadratic loss in (2.3) is small enough (cf. Goodfellow et al. (2016)). An implementation of the improvement would be to

(i) simulate n random inputs $\zeta_1, \cdots, \zeta_n$;

(ii) obtain the sample maximum $Y^{(i)}_{(n)} := \max\{G^{(i)}(\zeta_1), \cdots, G^{(i)}(\zeta_n)\}$ for every candidate $G^{(i)}$, $i = 1, \cdots, M$;

(iii) report $Y^*_{(n)} := \max\{Y^{(1)}_{(n)}, \cdots, Y^{(M)}_{(n)}\}$ as an estimate of the maximum of g.

This implementation requires about M times the computational costs of maximizing one neural network. The convergence rate of this implementation will be that of the most favourable candidate. For the polynomial distribution example at the beginning of this section, the illustrations in subsection 4.1 suggests the efficiency of the proposed improvement. As long as the distribution of one of the approximating neural networks has a lower root multiplicity at its maximum, the sample size needed in this example to reach a certain accuracy will be reduced by several magnitudes.

Examples 4.1-4.3 compare convergence rates of sample maxima from different neural networks along the same path, by inverting the output distributions. Let $\{U_n\}_n$ be a sequence of independent Uniform $[0, 1]$ samples. For a cumulative distribution function F, its inversion F^{-1} applied to every U_n has the distribution F. For different output distributions from different neural networks, the inverted distributions applied to the uniform samples can be viewed as the outputs of these neural networks with random inputs along the same path.

Example 4.1 *Let us suppose that the neural network output has the distribution F_r defined in* (4.2)*, with $y^* = 3$ and $y_* = 1$. Figure 4 shows the sample maximum $Y_{(n)} := \max\{F_r^{-1}(U_1), \cdots, F_r^{-1}(U_n)\}$, for different $r = 0, 1, 2, 3$. In this example, a lower root multiplicity r indeed speeds up the convergence.*

Example 4.2 *The coefficients of the output distribution is determined by the coefficients of the neural network (c.f. Abbas-Turki et al. (2023)). This example takes the distribution F_r in* (4.2) *to explore how the perturbation of its coefficients impacts the convergence rate. Since Example 4.1 has discussed the effect of the root multiplicity r, the focus here is y_* and y^* with $r = 1$. Let us suppose that the targeted distribution is F_r with $y^* = 3$ and $y_* = 1$.*

For the original distribution F_r, the sample maximum $Y_{(n)}$ is defined the same as in Example 4.1. To perturb the original distribution, we first slightly change y^* in the expression of F_r into 3.01 or 2.98 and respectively get the simulated sequences $\{Y_n(3.01)\}_n$ and $\{Y_n(2.98)\}_n$ by applying the inverted distributions to $\{U_n\}_n$. Let $\{Y_{(n)}(3.01)\}_n$ and $\{Y_{(n)}(2.98)\}_n$ denote the respective sample maxima of the two sequences. Then, for independent Uniform $[-1, 1]$ samples $\{\bar{U}_i\}_{i=1}^{100}$ and $\{\underline{U}_i\}_{i=1}^{100}$ which are also independent of $\{U_n\}_n$, we

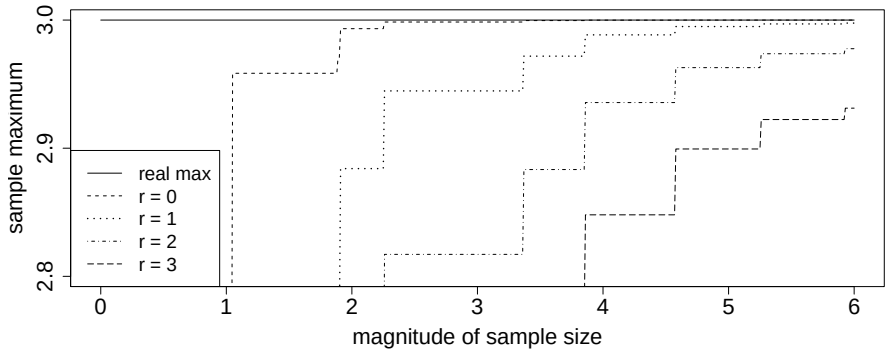

Figure 4: Sample maxima for $r = 0, 1, 2, 3$ along the same path, values below 2.8 not displayed.

generate the perturbed coefficients $\bar{y}_i := y^* + 0.01(y^* - y_*)\bar{U}_i$ *and* $\underline{y}_i = y_* + 0.01(y^* - y_*)\underline{U}_i$, *for* $i = 1, \cdots, 100$. *The perturbed distribution* $F_r^{(i)}$ *is produced from setting* $y^* = \bar{y}_i$ *and* $y_* = \underline{y}_i$ *in the targeted density* F_r. *Let* $Y_n^{(i)}$ *be the inversion of the perturbed distribution* $F_r^{(i)}$ *applied to the uniform random variable* U_n. *We take the sample maxima* $Y_{(n)}^{(i)} := \max\{Y_1^{(i)}, \cdots, Y_n^{(i)}\}$ *for* $i = 1, \cdots, 100$, *and the max* $Y_{(n)}^* := \max\{Y_{(n)}^{(1)}, \cdots, Y_{(n)}^{(100)}\}$ *of these sample maxima.*

Figure 5 illustrates sample maxima from the different versions, where in the legend "$(1, 3)$*", "*$(1, 3.01)$*", "*$(1, 2.98)$*" and "max of max" respectively represent the sequences* $\{Y_{(n)}\}_n$, $\{Y_{(n)}(3.01)\}_n$, $\{Y_{(n)}(2.98)\}_n$ *and* $\{Y_{(n)}^*\}_n$. *If willing to accept a bias, the perturbed version may converge faster, especially the maximum of the sample maxima from several randomly perturbed versions.*

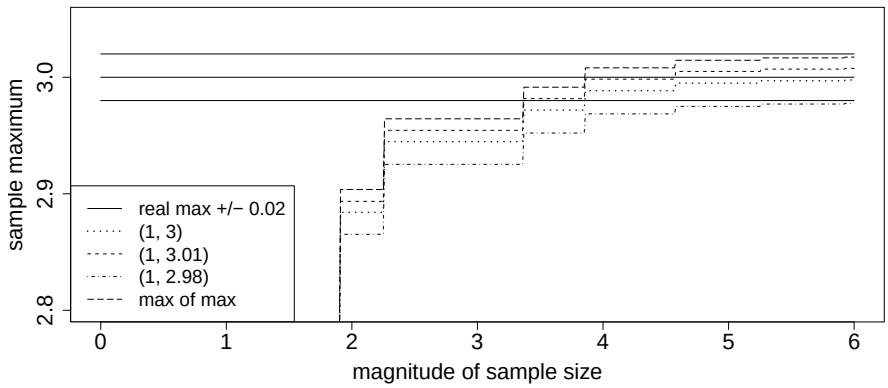

Figure 5: Sample maxima for $r = 1$ and different values of (y_*, y^*) along the same path, values below 2.8 not displayed.

Example 4.3 *Theorem 3.2 and Theorem 3.5 suggest that the limiting behaviors of the sample maximum* $Y_{(n)}$ *is determined by the root multiplicity of the survival function* $1 - F(y)$ *at the maximum* y^*. *In fact, what matters to the convergence rate is not only the root multiplicity, but also the shape of the tail. The reason is that, for a piecewise polynomial distribution, it might take too many samples before getting a sample near the maximum.*

For example, let us compare two probability densities. One density $f(y)$, *plotted as Figure 6 (right), is proportional to* $(y - 2.85)^2(y - 3)^2(y - 3.15)^2$ *on* $y \in [1, 3.15]$, *and is zero elsewhere. The other densities are* f_r *as in* (4.1) *with* $r = 6$, $y_* = 1$, *and* $y^* = 3, 3.15, 2.85$ *respectively. Figure 6 (left) shows the density* f_r *with* $r = 6$, $y_* = 1$, *and* $y^* = 3$. *Despite of the different root multiplicities of their corresponding survival functions, the two densities in Figure 6 look barely distinguishable.*

Figure 7 illustrates convergence rates of the sample maxima from the above distributions. We observe that the two densities in Figure 6 produce sample maxima ("$(1, 3)$" and "different roots") with similar convergence rates. However, different values of the coefficient y^* has a more obvious impact on the convergence rate.

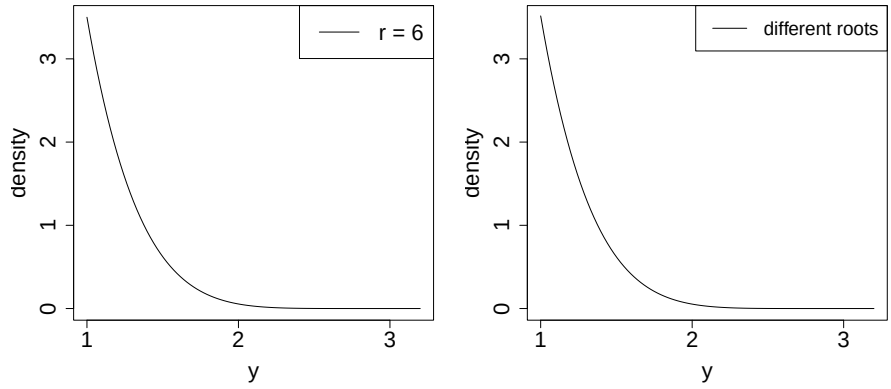

Figure 6: Density f_r for $r = 6$ (left) and the density with different roots (right).

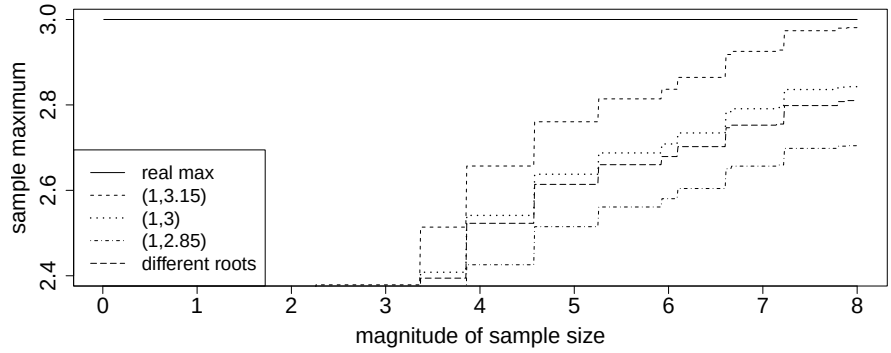

Figure 7: Sample maxima from different distributions, values below 2.4 not displayed.

5 Appendix: Proofs in section 2

Proof of Property 2.1. The neural network G is a composition of linear transformations and a continuous activation function, thus being continuous. A continuous function on a compact set achieves its maximum and minimum. □

Proof of Example 2.3. We may compute to get $F(y) = P(g(\zeta) \leq y) = 1 - \exp(-y/(1-y))$, for $0 \leq y < 1$.

(i) This statement follows from Definition 3.3.18, Example 3.3.22 and Proposition 3.3.25 on page 138-141 of Embrechts et al. (1997).

(ii) Using the distribution of $Y_{(n)}$ to express $\mathbb{E}\left[R_n^2\right]$ and applying a variable substitution in the integral, the mean-square error has the expression

$$\mathbb{E}\left[R_n^2\right] = n \int_0^1 (1-x)^{n-1}/(\ln x - 1)^2 dx \tag{5.1}$$

For any $\epsilon \in (0,1)$, because $\lim_{x\to 0+} x^{\epsilon}(\ln x - 1)^2 = 0$, there exists a positive constant $C(\epsilon)$, such that $0 \le x^{\epsilon}(\ln x - 1)^2 \le C(\epsilon)$ for $x \in [0,1]$. Then from (5.1) we may derive

$$\mathbb{E}\left[R_n^2\right] \ge \frac{n}{C(\epsilon)} \int_0^1 x^{-\epsilon}(1-x)^{n-1} dx = \frac{n}{C(\epsilon)} Beta(1-\epsilon, n). \tag{5.2}$$

□

Proof of Example 2.2. By Assumption 2.1 (ii), the neural network has the minimum value $y_* = a(0)$ and maximum $y^* = a(d)$.

For the random variable $\zeta = (\zeta_1, \cdots, \zeta_{d_0})$ uniformly distributed on the d_0-dimensional unit cube, its components are independent random variables uniformly distributed on [0,1]. The random variable $X := \sum_{k=1}^{d} \zeta_k$ is the sum of d independent Uniform [0,1] random variables, having an Irwin-Hall distribution (c.f. page 296 of Johnson et al. (1995)). The Irwin-Hall cumulative distribution function F_d is piecewise polynomial with degree at most d. Especially when $d-1 \le y \le d$, we may show by geometric method that the Irwin-Hall density equals

$$f_d(y) := \frac{1}{(d-1)!}(d-y)^{d-1}. \tag{5.3}$$

The corresponding tail probability is

$$1 - F_d(y) = \frac{1}{d!}(d-y)^d, \text{ for } d-1 \le y \le d. \tag{5.4}$$

We claim the existence of two positive constants $\underline{c} \le \bar{c}$, such that

$$\underline{c}(d-x) \le y^* - a(x) \le \bar{c}(d-x), \text{ for all } 0 \le x \le d. \tag{5.5}$$

In fact, the constants can be chosen as $\underline{c} = \bar{c} = a_+$ if the activation function is Leaky ReLU, and as

$$\underline{c} = \min\{a'(x) | 0 \le x \le d\} \text{ and } \bar{c} = \max\{a'(x) | 0 \le x \le d\} \tag{5.6}$$

if Assumption 2.1 is satisfied.

(i) Let F be the cummulative distribution function of $G(\zeta) = a(X)$. For $d-1 \le y \le d$, the tail probability is

$$1 - F(y) = \frac{1}{d!}(d - a^{-1}(y))^d. \tag{5.7}$$

Equation (5.4) and the two inequalities in (5.5) imply that

$$\frac{1}{d!\bar{c}^d}(y^* - y)^d \le 1 - F(y) \le \frac{1}{d!\underline{c}^d}(y^* - y)^d, \text{ for } d-1 \le y \le d. \tag{5.8}$$

Hence Assumption 3.3 (i) is satisfied with $\alpha = d$. The asymptotic distribution follows from Theorem 3.2.

Combining (5.7) with (3.2), and solving for $\frac{1}{d!}(d - a^{-1}(a(d) - c_n))^d = \frac{1}{n}$, we get

$$c_n = a(d) - a\left(d - (d!)^{1/d} n^{-1/d}\right). \tag{5.9}$$

By Assumption 2.1 (iii), Taylor's expansion can be applied to the function $a(d) - a(d-x)$ at $x = 0$ to get

$$c_n = a'(d)(d!)^{1/d} n^{-1/d} + o\left(n^{-1/d}\right). \tag{5.10}$$

The smaller order term in (5.10) may be omitted.

(ii) The second inequality in (5.5) implies that

$$\mathbb{E}\left[R_n^2\right] \le \bar{c}^2 \mathbb{E}\left[\left(d - X_{(n)}\right)^2\right], \tag{5.11}$$

where $X_{(n)}$ is the sample maximum of n independent realizations of X and the second expectation is its mean-square error. The distribution of X satisfies Assumption 3.3 (ii) with $\alpha = d$ and $\epsilon = 1$. Then Theorem 3.5 applies to the estimation of $\mathbb{E}\left[\left(d - X_{(n)}\right)^2\right]$. □

6 Appendix: Proofs in section 3

6.1 Proofs in subsection 3.1

Proof of Theorem 3.2. (i) This statement can be derived from Proposition 3.3.14 and Example 3.3.16 on page 136-137 of Embrechts et al. (1997).

(ii) Since $1 - F(y^*) = 0$ by Assumption 3.1, y^* is a root of the polynomial $1 - F(y)$. Under Assumption 3.3 (ii), the multiplicity α of the root y^* is at most d, which implies Assumption 3.3 (i). We may conclude statement (ii) from statement (i) of this theorem. □

Proof of Corollary 3.1. Because the sequence (2.13) converges in distribution to Weibull with parameter $\alpha > 0$, we have the expression

$$\lim_{n\to\infty} \mathbb{P}\left(\Psi_\alpha^{-1}(1-\lambda) \le c_n^{-1}(Y_{(n)} - y^*) \le 0\right) = \lambda, \tag{6.1}$$

which is equivalent to

$$\lim_{n\to\infty} \mathbb{P}\left(Y_{(n)} \le y^* \le Y_{(n)} - c_n\Psi_\alpha^{-1}(1-\lambda)\right) = \lambda. \tag{6.2}$$

□

6.2 Proofs in subsection 3.2

Let $\{U_i\}_{i=1}^\infty$ be a sequence of independent and identically distributed Uniform random variables on [0,1]. The sequences $\{Y_i\}_{i=1}^\infty$ and $\{U_i\}_{i=1}^\infty$ are independent of each other. With the quantile function Q defined in (3.6), the two sequences $\{Q(U_i)\}_{i=1}^\infty$ and $\{Y_i\}_{i=1}^\infty$ have the same distribution. The sample maximum $Y_{(n)}$ has the same distribution as $Q(U_{(n)})$, where $U_{(n)}$ is the maximum among $\{U_i\}_{i=1}^n$. The probability density function of $U_{(n)}$ is

$$f_{(n)}(u) = nu^{n-1}\mathbf{1}_{[0,1]}(u). \tag{6.3}$$

Proofs for Theorem 3.3 and Theorem 3.4 will all need to operate with the random variable $Q(U_{(n)})$ instead of the sample maximum $Y_{(n)}$.

Proof of Theorem 3.3. Let the function H be defined as

$$H(u) := Q(u) - y^*, \tag{6.4}$$

for $u \in [0,1]$. The difference R_n can be expressed as

$$R_n = Y_{(n)} - y^* \overset{\mathscr{D}}{=} Q(U_{(n)}) - y^* = H(U_{(n)}). \tag{6.5}$$

We then proceed to estimate the growth rate of $\mathbb{E}\left[R_n^2\right]$. Let $\{\epsilon_n\}_{n=1}^\infty$ be a decreasing sequence of positive real numbers valued in $(0,1)$ and converging to zero. Then $\mathbb{E}\left[R_n^2\right]$ is expressed as

$$\begin{aligned}\mathbb{E}\left[R_n^2\right] &= \int_0^1 H^2(u) f_{(n)}(u)du \\ &= \int_0^{1-\epsilon_n} H^2(u) f_{(n)}(u)du + \int_{1-\epsilon_n}^1 H^2(u) f_{(n)}(u)du.\end{aligned} \tag{6.6}$$

Let

$$H_{n,max} := \sup_{u\in(1-\epsilon_n,1]} |H(u)| \tag{6.7}$$

denote the supremum of H over $(1-\epsilon_n, 1]$. Because the quantile function Q is increasing, it holds that

$$H_{n,max} = y^* - Q(1-\epsilon_n). \tag{6.8}$$

By equations (6.3) and (6.6), the expectation $\mathbb{E}\left[R_n^2\right]$ is bounded by

$$\begin{aligned}&\mathbb{E}\left[R_n^2\right] \\ \le& \left(\sup_{u\in(0,1-\epsilon_n)} f_{(n)}(u)\right)\int_0^{1-\epsilon_n} H^2(u)du + H_{n,max}^2\int_{1-\epsilon_n}^1 f_{(n)}(u)du \\ =& n(1-\epsilon_n)^{n-1}\int_0^{1-\epsilon_n} H^2(u)du + H_{n,max}^2\left(1-(1-\epsilon_n)^n\right) \\ \le& n(1-\epsilon_n)^{n-1}\int_0^{1-\epsilon_n} H^2(u)du + H_{n,max}^2.\end{aligned} \tag{6.9}$$

By the definition of the quantile function Q in equation (3.6) and by the definition of H in equation (6.4), the integral $\int_0^{1-\epsilon_n} H^2(u)du$ is bounded by

$$\int_0^{1-\epsilon_n} H^2(u)du \leq (1-\epsilon_n)(y^* - y_*)^2. \tag{6.10}$$

The expressions (6.8), (6.9) and (6.10) imply that

$$\mathbb{E}\left[R_n^2\right] \leq n(1-\epsilon_n)^n (y^* - y_*)^2 + \left(y^* - Q(1-\epsilon_n)\right)^2. \tag{6.11}$$

Because the function

$$h(x) = \left(1 - \frac{1}{x}\right)^{-x}, \text{ for } x > 1, \tag{6.12}$$

converges to e from above, as $x \to +\infty$, the sequence

$$(1-\epsilon_n)^{-1/\epsilon_n} \tag{6.13}$$

converges to e from above, as $n \to \infty$. Hence we have

$$n(1-\epsilon_n)^n = n(1-\epsilon_n)^{(-1/\epsilon_n)(-\epsilon_n)n} \leq ne^{-n\epsilon_n}, \tag{6.14}$$

for any positive integer $n > 1$. In particular, we are allowed to take $\epsilon_n = n^{-\delta}$, for an arbitrary $\delta \in (0,1)$. Then the growth rate $ne^{-n\epsilon_n}$ in inequality (6.14) equals $ne^{-n^{1-\delta}}$. With this particular choice of ϵ_n, inequalities (6.11) and (6.14) imply the inequalities (3.7) and (3.8). □

Proof of Proposition 3.1: There exist two positive constants $\underline{c} \leq \bar{c}$ and a positive number Δy, such that

$$\underline{c}(y^* - y)^\alpha \leq 1 - F(y) \leq \bar{c}(y^* - y)^\alpha, \text{ for } y^* - \Delta y \leq y \leq y^*. \tag{6.15}$$

The inequalities in (6.15) are equivalent to

$$\bar{c}^{-1/\alpha}(1-F(y))^{-1/\alpha} \leq y^* - y \leq \underline{c}^{-1/\alpha}(1-F(y))^{-1/\alpha}, \text{ for } y^* - \Delta y \leq y \leq y^*. \tag{6.16}$$

The quantile Q is the inverse function of the cumulative distribution F, where F is continuous. Because F is assumed continuous in a neighbourhood of y^*, for n large enough, the quantile $Q\left(1-n^{-\delta}\right)$ is the value y such that the identity $1 - F(y) = n^{-\delta}$ holds. Then the inequalities in (6.16) imply that

$$\bar{c}^{-1/\alpha} n^{-\delta/\alpha} \leq y^* - Q\left(1-n^{-\delta}\right) \leq \underline{c}^{-1/\alpha} n^{-\delta/\alpha}, \text{ for } n \geq (1 - F(y^* - \Delta y))^{-1/\delta}. \tag{6.17}$$

For any real number $\delta \in (0,1)$, the sequence $\left\{ne^{-n^{1-\delta}}\right\}_{n=1}^{\infty}$ converges to zero faster than the sequence $\left\{n^{-2\delta/\alpha}\right\}_{n=1}^{\infty}$. Hence the dominating part in the convergence rate in inequality (3.8) is the second term. □

Proof of Theorem 3.4. By Assumption 3.4, the function $f(y)$ is neither zero nor infinity for y in a neighborhood $(y^* - \epsilon_0, y^*]$, for some $\epsilon_0 \in (0, \epsilon_F]$. Hence the cumulative distribution function F is strictly increasing and has an inverse function $F^{-1}(u) = Q(u)$ for $u \in (F(y^* - \epsilon_0), 1]$. The first and second order derivatives of $Q(u)$ for $u \in (F(y^* - \epsilon_0), 1]$ are

$$Q'(u) = \frac{1}{f(Q(u))};\ Q''(u) = -\frac{f(Q(u))}{f^3(Q(u))}. \tag{6.18}$$

The function H is defined as

$$H(u) := Q(u) - Q(1) - Q'(1)(u-1) = Q(u) - y^* - \frac{1}{f(y^*)}(u-1). \tag{6.19}$$

Then the difference R_n can be expressed as

$$\begin{aligned} R_n = Y_{(n)} - y^* &\overset{\mathscr{D}}{=} Q(U_{(n)}) - Q(1) \\ &= Q'(1)(U_{(n)} - 1) + H(U_{(n)}) \\ &= \frac{1}{f(y^*)}(U_{(n)} - 1) + H(U_{(n)}). \end{aligned} \tag{6.20}$$

The mean-square error $\mathbb{E}\left[R_n^2\right]$ is bounded by

$$\mathbb{E}\left[R_n^2\right] \leq 2\left(\frac{1}{f^2(y^*)}\mathbb{E}\left[\left(U_{(n)}-1\right)^2\right]+\mathbb{E}\left[H^2(U_{(n)})\right]\right). \tag{6.21}$$

To estimate the convergence rate of $\mathbb{E}\left[R_n^2\right]$, we shall respectively derive upper bounds of the two quantities $\mathbb{E}\left[\left(U_{(n)}-1\right)^2\right]$ and $\mathbb{E}\left[H^2(U_{(n)})\right]$.

By equation (6.3), we may compute to get

$$\begin{aligned}&\mathbb{E}\left[\left(U_{(n)}-1\right)^2\right]=\mathbb{E}\left[U_{(n)}^2\right]-2\mathbb{E}\left[U_{(n)}\right]+1\\ =&\int_0^1 nu^{n+1}du-2\int_0^1 nu^n du+1=\frac{2}{(n+1)(n+2)}.\end{aligned} \tag{6.22}$$

We then proceed to estimate the growth rate of $\mathbb{E}\left[H^2(U_{(n)})\right]$. Let $\{\epsilon_n\}_{n=1}^{\infty}$ be a sequence of positive real numbers valued in $(0,1)$ and converging to zero. Then $\mathbb{E}\left[H^2(U_{(n)})\right]$ is expressed as

$$\begin{aligned}&\mathbb{E}\left[H^2(U_{(n)})\right]=\int_0^1 H^2(u)f_{(n)}(u)du\\ =&\int_0^{1-\epsilon_n} H^2(u)f_{(n)}(u)du+\int_{1-\epsilon_n}^1 H^2(u)f_{(n)}(u)du.\end{aligned} \tag{6.23}$$

Let

$$H_{n,max}:=\sup_{u\in(1-\epsilon_n,1]}|H(u)| \tag{6.24}$$

denote the supremum of H over $(1-\epsilon_n,1]$. By equations (6.3) and (6.23), the expectation $\mathbb{E}\left[H^2(U_{(n)})\right]$ is bounded by

$$\begin{aligned}&\mathbb{E}\left[H^2(U_{(n)})\right]\\ \leq&\left(\sup_{u\in(0,1-\epsilon_n)}f_{(n)}(u)\right)\int_0^{1-\epsilon_n}H^2(u)du+H_{n,max}^2\int_{1-\epsilon_n}^1 f_{(n)}(u)du\\ =&n(1-\epsilon_n)^{n-1}\int_0^{1-\epsilon_n}H^2(u)du+H_{n,max}^2\left(1-(1-\epsilon_n)^n\right)\\ \leq&n(1-\epsilon_n)^{n-1}\int_0^{1-\epsilon_n}H^2(u)du+H_{n,max}^2.\end{aligned} \tag{6.25}$$

By equation (6.19), the value of $H(u)$ is bounded

$$y_*-y^*\leq Q(u)-y^*\leq H(u)\leq\frac{1}{f(y^*)}(1-u)\leq\frac{1}{f(y^*)}, \tag{6.26}$$

for any $u\in[0,1]$. It follows that

$$\int_0^{1-\epsilon_n}H^2(u)du\leq(1-\epsilon_n)\left((y^*-y_*)^2+\frac{1}{f^2(y^*)}\right). \tag{6.27}$$

By equations (6.18) and (6.19) and by the Taylor Expansion Theorem, for every $u\in(F(y^*-\epsilon_0),1)$, there exists a real number $u_0\in[u,1)$, such that

$$H(u)=\frac{1}{2}Q''(u_0)(u-1)^2=\frac{1}{2}\left(\frac{1}{f(Q(u_0))}\right)'(u-1)^2=-\frac{f'(Q(u_0))}{2f^3(Q(u_0))}(u-1)^2. \tag{6.28}$$

When

$$\epsilon_n<1-F(y^*-\epsilon_0), \tag{6.29}$$

the inverval $(1-\epsilon_n,1]$ is strictly included in the neighborhood $(F(y^*-\epsilon_0),1]$ where Q'' has the explicit expression (6.18). It follows from equations (6.24) and (6.28), as well as Assumption 3.4, that

$$H_{n,max}^2\leq\frac{1}{4}\left(\sup_{y\in(Q(1-\epsilon_n),y^*]}\left\{\frac{(f'(y))^2}{f^6(y)}\right\}\right)\epsilon_n^4. \tag{6.30}$$

The expressions (6.25), (6.27) and (6.30) imply that

$$\mathbb{E}\left[H^2(U_{(n)})\right] \leq \left((y^*-y_*)^2+\frac{1}{f^2(y^*)}\right)n(1-\epsilon_n)^n+\frac{1}{4}\left(\sup_{y\in(Q(1-\epsilon_n),y^*]}\left\{\frac{(f'(y))^2}{f^6(y)}\right\}\right)\epsilon_n^4. \tag{6.31}$$

In particular, we may take $\epsilon_n = n^{-\delta}$, for an arbitrary $\delta \in (0,1)$. Inequality (6.29) is equivalent to

$$n > (1-F(y^*-\epsilon_0))^{-\delta}. \tag{6.32}$$

The estimates (6.21), (6.22), (6.31), (6.32) and (6.14) conclude inequality (3.10). The inequality (3.10) implies that, as $n\to\infty$, $\mathbb{E}\left[R_n^2\right]$ is has the convergence rate

$$\mathbb{E}\left[R_n^2\right] \leq O(n^{-2})+O(ne^{-n^{1-\delta}})+O(n^{-4\delta}). \tag{6.33}$$

If further requiring that $\delta \in \left(\frac{1}{2},1\right)$, we get inequality (3.11) from inequality (6.33). □

Proof of Theorem 3.5. Conditioning on whether $Y_{(n)} \leq y^*-\epsilon$ or not, the mean-square error can be expressed as

$$\mathbb{E}\left[R_n^2\right] = \mathbb{E}\left[R_n^2\middle| Y_{(n)} \leq y^*-\epsilon\right](F(y^*-\epsilon))^n + \mathbb{E}\left[R_n^2\middle| Y_{(n)} > y^*-\epsilon\right](1-(F(y^*-\epsilon))^n), \tag{6.34}$$

where the two summands are respectively bounded from above by $(y^*-y_*)^2(F(y^*-\epsilon))^n$ and $\mathbb{E}\left[R_n^2\middle| Y_{(n)} > y^*-\epsilon\right]$. It remains to estimate the second summand.

Under Assumption 3.3 (ii), the function q defined as $q(y) := (1-F(y))(y^*-y)^{-\alpha}$ is a polynomial of degree $d-\alpha$, satisfying $q(y^*)>0$ and $0<q(y)(y^*-y)^\alpha \leq 1$ for $y^*-\epsilon \leq y < y^*$. Then for $y^*-\epsilon \leq y \leq y^*$, the cumulative distribution function and the density can be expressed as

$$F(y) = 1-q(y)(y^*-y)^\alpha \text{ and } f(y) = (\alpha q(y)-q'(y)(y^*-y))(y^*-y)^{\alpha-1}. \tag{6.35}$$

The constant

$$C_1(\epsilon) := \min\{q(y)|y^*-\epsilon \leq y \leq y^*\} \tag{6.36}$$

satisfy $C_1(\epsilon)>0$, $C_1(\epsilon)(y^*-y)^\alpha \leq q(y)(y^*-y)^\alpha \leq 1$ and thus $F(y) \leq 1-C_1(\epsilon)(y^*-y)^\alpha$, for any $y^*-\epsilon \leq y < y^*$. Especially with $y = y^*-\epsilon$, we know that $0<C_1(\epsilon)\epsilon^\alpha \leq 1$. Because y^* is the maximum, the density $f(y)$ cannot be all-zero on $[y^*-\epsilon, y^*)$ and the constant

$$C_2(\epsilon) := \max\{\alpha q(y)-q'(y)(y^*-y)|y^*-\epsilon \leq y \leq y^*\} \tag{6.37}$$

thus satisfies $f(y) \leq C_2(\epsilon)(y^*-y)^{\alpha-1}$. For $y^*-\epsilon \leq y \leq y^*$, we have the estimate

$$f_{(n)}(y) = n(F(y))^{n-1}f(y) \leq C_2(\epsilon)n(1-C_1(\epsilon)(y^*-y)^\alpha)^{n-1}(y^*-y)^{\alpha-1}. \tag{6.38}$$

From the conditional probability

$$\mathbb{P}(Y_{(n)} \leq y | Y_{(n)} > y^*-\epsilon) = (F_{(n)}(y)-F_{(n)}(y^*-\epsilon))/(1-F_{(n)}(y^*-\epsilon)), \tag{6.39}$$

we may get the conditional density and its upper bound

$$\begin{aligned} &f_{(n)}(y|Y_{(n)} > y^*-\epsilon) = f_{(n)}(y)/(1-(F(y^*-\epsilon))^n) \\ \leq &\frac{C_2(\epsilon)}{1-F(y^*-\epsilon)}n(1-C_1(\epsilon)(y^*-y)^\alpha)^{n-1}(y^*-y)^{\alpha-1}. \end{aligned} \tag{6.40}$$

Hence the conditional mean-square error

$$\mathbb{E}\left[R_n^2\middle| Y_{(n)} > y^*-\epsilon\right] = \int_{y^*-\epsilon}^{y^*}(y^*-y)^2 f_{(n)}(y|Y_{(n)} > y^*-\epsilon)dy \tag{6.41}$$

has the upper bound

$$\mathbb{E}\left[R_n^2\middle| Y_{(n)} > y^*-\epsilon\right] \leq \frac{C_2(\epsilon)}{1-F(y^*-\epsilon)}n\int_{y^*-\epsilon}^{y^*}(1-C_1(\epsilon)(y^*-y)^\alpha)^{n-1}(y^*-y)^{\alpha+1}dy. \tag{6.42}$$

With the variable substitution $x = C_1(\epsilon)(y^*-y)^\alpha$ and the notation $C(\epsilon,\alpha) := C_2(\epsilon)/(\alpha C_1(\epsilon)^{1+2/\alpha}(1-F(y^*-\epsilon)))$ on its right hand side, the inequality (6.42) becomes

$$\mathbb{E}\left[R_n^2\middle| Y_{(n)} > y^*-\epsilon\right] \leq C(\epsilon,\alpha)n\int_0^{C_1(\epsilon)\epsilon^\alpha}(1-x)^{n-1}x^{2/\alpha}dx. \tag{6.43}$$

Because $0<C_1(\epsilon)\epsilon^\alpha \leq 1$ and in Assumption 3.3 (ii) the number ϵ can be taken within $(0,1]$, the integral on the right hand side of (6.43) is bounded from above by $Beta(1+2/\alpha, n)$. □

References

Abbas-Turki, L. A., B. Alexandrine, and Q. Li (2023). Polynomial distribution of feedforward neural network output. *Preprint*.

Abbas-Turki, L. A., S. Crépey, and B. Saadeddine (2023). Pathwise CVA regressions with oversimulated defaults. *Mathematical Finance 33*(2), 274–307.

Bachouch, A., C. Huré, N. Langrené, and H. Pham (2022). Deep neural networks algorithms for stochastic control problems on finite horizon: numerical applications. *Methodology and Computing in Applied Probability 24*(1), 143–178.

Bahadur, R. R. (1966). A note on quantiles in large samples. *The Annals of Mathematical Statistics 37*(3), 577–580.

Bertsekas, D. (2012). *Dynamic programming and optimal control: Volume I*, Volume 4. Athena scientific.

Broadie, M., Y. Du, and C. C. Moallemi (2015). Risk estimation via regression. *Operations Research 63*(5), 1077–1097.

Castelló, A., S. Barrachina, M. F. Dolz, E. S. Quintana-Ortí, P. S. Juan, and A. E. Tomás (2022). High performance and energy efficient inference for deep learning on multicore ARM processors using general optimization techniques and BLIS. *Journal of Systems Architecture 125*, 102459.

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

Duttweiler, D. L. (1973). The mean-square error of Bahadur's order-statistics approximation. *The Annals of Statistics 1*(3), 446–453.

Embrechts, P., C. Klüppelberg, and T. Mikosch (1997). *Modelling Extremal Events for Insurance and Finance*. Springer-Verlag.

Gobet, E., J.-P. Lemor, and X. Warin (2005). A regression-based monte carlo method to solve backward stochastic differential equations. *The Annals of Applied Probability 15*(3), 2172–2202.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. The MIT Press.

Hamacher, K. (2006). Adaptation in stochastic tunneling global optimization of complex potential energy landscapes. *Europhysics Letters 74*(6), 944.

Hansen, N. (2006). The CMA evolution strategy: a comparing review. *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, 75–102.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks 4*(2), 251–257.

Huré, C., H. Pham, A. Bachouch, and N. Langrené (2021). Deep neural networks algorithms for stochastic control problems on finite horizon: Convergence analysis. *SIAM Journal on Numerical Analysis 59*(1), 525–557.

Johnson, N. L., S. Kotz, and N. Balakrishnan (1995). *Continuous Univariate Distributions* (2nd ed.), Volume 2. John Wiley & Sons.

Kidger, P. and T. Lyons (2020). Universal Approximation with Deep Narrow Networks. In J. Abernethy and S. Agarwal (Eds.), *Proceedings of Thirty Third Conference on Learning Theory*, Volume 125 of *Proceedings of Machine Learning Research*, pp. 2306–2327. PMLR.

Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Marinari, E. and G. Parisi (1992). Simulated tempering: a new Monte Carlo scheme. *Europhysics letters 19*(6), 451.

Mishchenko, K., F. F. Bach, M. Even, and B. Woodworth (2022). Asynchronous sgd beats minibatch sgd under arbitrary delays. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), *Advances in Neural Information Processing Systems*, Volume 35, pp. 420–433. Curran Associates, Inc.

Schulman, J., F. Wolski, P. Dhariwal, A. Radford, and O. Klimov (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Shi, S., Q. Wang, P. Xu, and X. Chu (2016). Benchmarking state-of-the-art deep learning software tools. pp. 99–104.

Sutton, R. S. and A. G. Barto (1998). *Reinforcement Learning: An Introduction*. The MIT Press.