Electronic Journal of Statistics
Vol. 18 (2024) 5601–5658
ISSN: 1935-7524
https://doi.org/10.1214/24-EJS2332

A supervised deep learning method for nonparametric density estimation

Thijs Bos[1] **and Johannes Schmidt-Hieber**[2]

[1] *Leiden University,*
e-mail: j.m.bos@math.leidenuniv.nl

[2] *University of Twente,*
e-mail: a.j.schmidt-hieber@utwente.nl

Abstract: Nonparametric density estimation is an unsupervised learning problem. In this work we propose a two-step procedure that casts the density estimation problem in the first step into a supervised regression problem. The advantage is that we can afterwards apply supervised learning methods. Compared to the standard nonparametric regression setting, the proposed procedure creates, however, dependence among the training samples. To derive statistical risk bounds, one can therefore not rely on the well-developed theory for i.i.d. data. To overcome this, we prove an oracle inequality for this specific form of data dependence. As an application, it is shown that under a compositional structure assumption on the underlying density, the proposed two-step method achieves convergence rates that are faster than the standard nonparametric rates. A simulation study illustrates the finite sample performance.

MSC2020 subject classifications: Primary 62G07; secondary 68T07.
Keywords and phrases: Neural networks, nonparametric density estimation, statistical estimation rates, (un)supervised learning.

Received May 2024.

Contents

arXiv: 2306.10471

1. Introduction

Machine learning distinguishes between supervised and unsupervised learning tasks [11, 50]. In the supervised framework, the dataset consists of input-output pairs. No outputs are observed in the unsupervised setting. For supervised learning, classical examples are regression and classification; for unsupervised learning, commonly encountered problems are density estimation and clustering. The apparent difference between supervised and unsupervised tasks resulted in machine learning methods that either apply to the supervised or to the unsupervised framework. While neural nets can be applied in both scenarios, the underlying methodology is mostly unrelated: In the supervised context, deep learning is applied to reconstruct the function mapping the inputs to the outputs; in the unsupervised framework, neural networks are employed for instance in ODE-based models for density estimation [29, 55, 47] or for feature extraction, e.g. by making use of variational autoencoders [38]. Moreover, generative AI methods such as generative adversarial networks (GANs) or diffusion models invoke neural networks and can be viewed as density estimators [21, 16, 64, 15, 69, 54].

While there has been previous work transforming density estimation into a binary classification problem, see Section 14.2.4 in [31] and [30], in this article, we show how unsupervised multivariate density estimation can be cast into a supervised regression problem. For that, we generate suitable response variables from the data in a first step. Rewriting the problem as supervised learning task allows us to borrow strength from supervised learning methods. We demonstrate this by fitting deep ReLU networks. In the theoretical deep learning literature, it has been shown that supervised deep networks can outperform other methods if the target function exhibits some compositional structure. Making the link to supervised learning allows us to exploit this property also for density estimation. This is highly desirable as a compositional structure is frequently imposed in modelling of densities. Examples include copula models [1, 51] and Bayesian network models [41], see also Section 4.

Theorem 3.1 is our main theoretical contribution and establishes an oracle inequality for supervised regression methods applied to nonparametric density estimation. The key technical difficulty in the proof is to deal with the dependence incurred by generating the response variables in the first step of the proposed method. To control the dependence, we use a Poissonization argument. Applying the derived oracle inequality, we show in Theorem 3.4 that deep ReLU networks can obtain fast convergence rates, given that the underlying density has a compositional structure. For sufficiently smooth densities, the convergence rates are, up to logarithmic factors in the sample size, the same as the recently obtained minimax rates in the nonparametric regression model under compositional structure on the regression function, [61]. But there are also smoothness regimes where the convergence rate is slower by a polynomial order in the sample size if compared to the nonparametric regression case. This is due to the first step in the construction of the estimator that transforms the density estimation problem into a supervised regression problem. But still then there are scenarios where the convergence rate is considerably faster than doing off-the-shelf kernel density estimation without taking the underlying compositional structure of the density into account.

The paper is structured as follows. Section 2 describes the construction of suitable response variables from the data. In Section 3 we present a suitable oracle inequality for non-i.i.d. data. Furthermore, we provide convergence rates in the case that the regression estimator is a deep neural network and the underlying density are compositional functions. In Section 4 we shortly discuss some density models that exhibit compositional structure. A small (exploratory) simulation is provided in Section 5. Section 6 summarizes related literature. Almost all proofs are deferred to the Appendix.

1.1. Notation

We denote vectors and vector valued functions by bold letters. For a vector $\mathbf{x} = (x_1, \dots, x_k)^\top$ we define $|\mathbf{x}|_\infty = \max_{i=1,\dots,k} |x_i|$, $|\mathbf{x}|_1 = \sum_{i=1}^k |x_i|$. and $|\mathbf{x}|_0 = \sum_{i=1}^k \mathbb{1}_{\{x_i \neq 0\}}$. For partial derivatives we use multi-index notation, that is, if $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_d) \in \{0, 1, 2, \dots\}^d$ we set $\partial^{\boldsymbol{\alpha}} := \partial_{x_1}^{\alpha_1} \dots \partial_{x_d}^{\alpha_d}$. We denote the supremum norm of a function $f : \mathcal{D} \to \mathbb{R}$ by $\|f\|_\infty = \sup_{\mathbf{x} \in \mathcal{D}} |f(\mathbf{x})|$. As commonly defined in nonparametric statistics, for a real number $x \in \mathbb{R}$, $\lfloor x \rfloor$ is the largest integer $< x$ and $\lceil x \rceil$ is the smallest integer $\geq x$. The minimum and maximum of two real numbers x, y are also denoted by the respective expressions $x \wedge y$ and $x \vee y$. For two sequences $(a_n)_n$ and $(b_n)_n$, we write $a_n \lesssim b_n$ if there exists a constant C such that $a_n \leq C b_n$ for all n. Moreover, $a_n \asymp b_n$ means that $a_n \lesssim b_n$ and $b_n \lesssim a_n$. If no basis is specified, then $\log = \ln$.

2. Conversion into a supervised regression problem

We consider nonparametric density estimation on the hypercube $[0, 1]^d$, where we observe $2n$ i.i.d. random vectors $\mathbf{X}_i \in [0, 1]^d$ which are distributed accord-

ing to an unknown density f_0 from a nonparametric class. The density estimation problem is to recover this density f_0 from the data $\mathbf{X}_1, \ldots, \mathbf{X}_{2n}$. Here the sample size $2n$ is chosen for notational convenience, as we will do data splitting. It is moreover convenient to rename the second half of the dataset and denote it by $\mathbf{X}'_1, \ldots, \mathbf{X}'_n$. Thus, we are observing the $2n$ i.i.d. random vectors $(\mathbf{X}_1, \ldots, \mathbf{X}_n, \mathbf{X}'_1, \ldots, \mathbf{X}'_n)$. In the first step of the proposed method, the second half of the sample $\mathbf{X}'_1, \ldots, \mathbf{X}'_n$ is used to compute an undersmoothed kernel density estimator. From that we construct a response variable Y_i for each of the remaining datapoints $\mathbf{X}_i$. The response variables Y_i can be interpreted as noisy versions of $f_0(\mathbf{X}_i)$. The augmented data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ are now viewed as a nonparametric regression problem with the unknown density f_0 as regression function. Thus, any nonparametric regression technique could be applied to recover the regression function f_0 from the supervised data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$. Here we propose to fit a deep neural network. This is motivated by previous work which has shown that deep neural networks can adapt to various forms of structural constraints and avoid the curse of dimensionality [39, 57, 7, 61, 40]. As we will argue below, such structural constraints occur in modelling of multivariate densities. Fitting a neural network to the regression data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ is therefore natural.

In nonparametric statistics, a function $K : \mathbb{R} \to \mathbb{R}$ is called a kernel if K is integrable with $\int K(u)\, du = 1$. If for some positive integer s, we moreover have vanishing moments $\int u^\ell K(u)\, du = 0$ for all $\ell = 1, \ldots, s$, and $\int |u|^{s+1} |K(u)|\, du < \infty$, then K is called a kernel of order s. We now outline the two steps of the method.

Step 1: Choose a kernel K with $\|K\|_\infty < \infty$ and support on $[-1, 1]$. For a bandwidth h_n satisfying $(\log(n)/n)^{1/d} \leq h_n \leq 2(\log(n)/n)^{1/d}$ and such that h_n^{-1} is a positive integer for all $n > 1$ (existence of such a sequence is guaranteed by Lemma 7.1), consider the multivariate kernel density estimator based on the subsample $\mathbf{X}'_1, \ldots, \mathbf{X}'_n$ with $\mathbf{X}'_\ell = (X'_{\ell,1}, \ldots, X'_{\ell,d})^\top$ given by

$$\widehat{f}_{\mathrm{KDE}}(\mathbf{x}) := \frac{1}{n h_n^d} \sum_{\ell=1}^{n} \prod_{r=1}^{d} K\left(\frac{X'_{\ell,r} - x_r}{h_n}\right), \tag{2.1}$$

and using the notation $\mathbf{x} = (x_1, \ldots, x_d)$. For $i = 1, \ldots, n$, define

$$Y_i := \widehat{f}_{\mathrm{KDE}}(\mathbf{X}_i). \tag{2.2}$$

Setting $\varepsilon_i := Y_i - f_0(\mathbf{X}_i)$, we obtain the regression model

$$Y_i = f_0(\mathbf{X}_i) + \varepsilon_i, \quad i = 1, \ldots, n. \tag{2.3}$$

Step 2: Compute an estimator $\widehat{f}$ based on the data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$.

Definition 2.1. We refer to any such $\widehat{f}$ as two-stage nonparametric density estimator. If the kernel is of order s, we call $\widehat{f}$ the two-stage nonparametric density estimator with kernel of order s.

Both $\widehat{f}$ and the kernel density estimator $\widehat{f}_{\mathrm{KDE}}$ are estimators for f_0. However, because of the small bandwidth, the kernel density estimator severely undersmooths. The variance of $\widehat{f}_{\mathrm{KDE}}(\mathbf{x})$ at a given point $\mathbf{x}$ is known to scale with $1/(nh_n^d) \asymp 1/\log(n)$. This means that the noise variables ε_i scale with $1/\sqrt{\log(n)}$ in the sample size. Therefore, the denoising happens in the second step of the proposed two-step procedure.

Although the notation seems to suggest that (2.3) is the standard nonparametric regression framework, all data points depend on the underlying kernel density estimator $\widehat{f}_{\mathrm{KDE}}$. The pairs $(\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n)$ are henceforth dependent and thus not i.i.d. To deal with this dependence is the main technical challenge in the analysis of the proposed method.

The kernel density estimator in Step 1 undersmooths and does not require knowledge of the true smoothness. The conditions on the kernel K are standard. Taking a kernel of order s together with an optimal bandwidth choice is known to lead to optimal convergence rates if the smoothness of the density is at most $s+1$. The fact that the bandwidth is chosen such that h_n^{-1} is a positive integer allows us to partition $[0,1]$ into h_n^{-1} disjoint intervals of length h_n.

For fitting a function to the data $(\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n)$ in the second step of the procedure, machine learning methods aim to minimize a loss. For regression, the most common choice is the least squares loss $\frac{1}{n}\sum_{i=1}^n \big(Y_i - f(\mathbf{X}_i)\big)^2$. The least squares estimator $\widehat{f}_n$ over a function class $\mathcal{F}$ for the density f_0 is defined as any global minimizer of the least squares loss

$$\widehat{f}_n \in \operatorname*{arg\,min}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \big(Y_i - f(\mathbf{X}_i)\big)^2.$$

Due to the nonconvex energy landscape, neural network training usually does not find the global minimum. The difference between training error of the estimator and training error of the global minimum is commonly referred to as optimization error. For any estimator $\widehat{f}$ taking values in a function class $\mathcal{F}$, and data generated from the nonparametric regression model with regression function f_0, we consider here the optimization error

$$\Delta_n(\widehat{f}, f_0) := \mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n (Y_i - \widehat{f}(\mathbf{X}_i))^2 - \inf_{f \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^n (Y_i - f(\mathbf{X}_i))^2\right], \tag{2.4}$$

where the expectation is taken over the full data set, making $\Delta_n(\widehat{f}, f_0)$ deterministic.

The risk of an estimator $\widetilde{f}$ is given by

$$R(\widetilde{f}, f_0) := \mathbb{E}_{f_0,\mathbf{X}}\left[(\widetilde{f}(\mathbf{X}) - f_0(\mathbf{X}))^2\right] = \int \mathbb{E}_{f_0}\left[(\widetilde{f}(\mathbf{x}) - f_0(\mathbf{x}))^2\right] f_0(\mathbf{x})\, d\mathbf{x}. \tag{2.5}$$

Here $\mathbf{X} \overset{d}{=} \mathbf{X}_1$ is independent of the data and $\mathbb{E}_{f_0,\mathbf{X}}$ is the expectation with respect to the joint distribution of $\mathbf{X}$ and the data set. We denote by $\mathbb{E}_{\mathbf{X}}$ the expectation with respect to $\mathbf{X}$.

3. Main results

We assume that the density f_0 belongs to the class of β-Hölder smooth functions on $\mathbb{R}^d$ with support on $[0,1]^d$. For $\beta > 0$ and a domain $\mathcal{D} \subseteq \mathbb{R}^d$, the ball of β-Hölder functions with radius Q is defined as

$$\mathcal{H}_d^\beta(\mathcal{D}, Q) := \Bigg\{ f : \mathcal{D} \to \mathbb{R} : \sum_{\boldsymbol{\gamma}:\, 0 \le |\boldsymbol{\gamma}|_1 < \beta} \|\partial^{\boldsymbol{\gamma}} f\|_\infty + \sum_{\boldsymbol{\gamma}: |\boldsymbol{\gamma}|_1 = \lfloor \beta \rfloor} \sup_{\mathbf{x},\mathbf{y} \in \mathcal{D}, \mathbf{x} \neq \mathbf{y}} \frac{|\partial^{\boldsymbol{\gamma}} f(\mathbf{x}) - \partial^{\boldsymbol{\gamma}} f(\mathbf{y})|}{|\mathbf{x} - \mathbf{y}|_\infty^{\beta - \lfloor \beta \rfloor}} \le Q \Bigg\}, \tag{3.1}$$

where $\|\cdot\|_\infty$ denotes the supremum norm on $\mathcal{D}$, $\partial^{\boldsymbol{\gamma}} = \partial_{x_1}^{\gamma_1} \dots \partial_{x_d}^{\gamma_d}$, and $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_d) \in \{0, 1, 2, \dots\}^d$. To define the partial derivatives if $\mathcal{D}$ is not an open set, we assume that there exists an open set $\mathcal{U} \supset \mathcal{D}$ and an extension of f on $\mathcal{U}$ for which the partial derivatives $\partial^{\boldsymbol{\gamma}}$ for all $\boldsymbol{\gamma}$ with $|\boldsymbol{\gamma}|_1 < \beta$ are defined. The class of β-Hölder smooth densities on $\mathbb{R}^d$ and support on $[0,1]^d$ can subsequently be defined as

$$\mathcal{C}_d^\beta(Q) := \left\{ f \in \mathcal{H}_d^\beta(\mathbb{R}^d, Q) : \operatorname{supp} f \subseteq [0,1]^d, \int_{[0,1]^d} f(\mathbf{x})\, d\mathbf{x} = 1, f \ge 0 \right\}.$$

We define the class of β-Hölder smooth densities on $[0,1]^d$ by restricting β-Hölder smooth densities on $\mathbb{R}^d$ to $[0,1]^d$,

$$\mathcal{C}_d^\beta([0,1]^d, Q) := \left\{ f|_{[0,1]^d} : f \in \mathcal{C}_d^\beta(Q) \right\}.$$

Below we assume that the true density lies in this space. The condition that densities in this space can be extended to smooth functions on $\mathbb{R}^d$ is imposed to avoid (technical) difficulties of the kernel density estimator near the boundary of $[0,1]^d$. For a reference dealing with the behaviour of kernel estimators near boundaries, see Section 2.11 of [74].

We state the oracle inequality for estimators taking values in an abstract function class $\mathcal{F}(F) \subseteq \{f : \|f\|_\infty \le F\}$. For that, we denote by $\mathcal{N}_{\mathcal{F}}(\delta)$ the covering number of a class $\mathcal{F}(F)$ with respect to the supremum norm. More specifically, $\mathcal{N}_{\mathcal{F}}(\delta)$ is the smallest number of supremum norm balls with radius δ and centers contained in $\mathcal{F}$ that are necessary to cover $\mathcal{F}$.

Theorem 3.1. *For $n \ge 3$, consider the density estimation model defined by (2.1)-(2.3) with density f_0 in the Hölder class $\mathcal{C}_d^\beta([0,1]^d, Q)$. Let $\widehat{f}$ be a two-stage nonparametric density estimator with kernel of order $\lfloor \beta \rfloor$ as defined in Definition 2.1. If $\widehat{f}$ takes values in the function class $\mathcal{F} = \mathcal{F}(F)$, with $F \ge \max\{Q, 1\}$, then there exist constants C_1, C_2, C_3 only depending on F, K, d, Q, β*

such that for any $\delta > 0$,

$$R(\widehat{f}, f_0) \leq C_1 \frac{\log^2(n)\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} + C_2\delta + C_3 \left(\frac{\log(n)}{n}\right)^{\frac{2\beta}{d}} + 6\Delta_n(\widehat{f}, f_0) + 7 \inf_{f \in \mathcal{F}} \mathbb{E}_{\mathbf{X}} \left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right].$$

As common for oracle inequalities, the upper bound contains an approximation term, a complexity term involving the metric entropy, and the optimization error $\Delta_n(\widehat{f}, f_0)$. For neural networks and other parametrizable function classes, the metric entropy $\log(\mathcal{N}_{\mathcal{F}}(\delta))$ depends only logarithmically on δ and one can choose $\delta = 1/n$, making the $C_2\delta$ term negligibly small.

If compared to oracle inequalities for i.i.d. data, the bound contains moreover the term $C_3(\log(n)/n)^{2\beta/d}$ that is due to the bandwidth choice $h_n \asymp (\log(n)/n)^{1/d}$ and a term of the order $h_n^{2\beta}$ that can be traced back to Proposition 7.3. To decrease the order of the $C_3(\log(n)/n)^{2\beta/d}$ term, it is tempting to aim for a smaller bandwidth $h_n \ll n^{-1/d}$. However, even if the data points are equally spaced in $[0,1]^d$, the distance of two neighboring data points is $n^{-1/d}$. Thus for bandwidth $h_n \ll n^{-1/d}$, it follows from the definition of the kernel density estimator in (2.1) that the estimated density degenerates into separate spikes centered around the data points $\mathbf{X}'_1, \ldots, \mathbf{X}'_n$. As a consequence, a generated response variables Y_i will likely either be extremely large or attain a value near zero and the two-step method that we propose will not work anymore.

Compared to oracle inequalities for i.i.d. data in the nonparametric regression model, the main difficulty in the proof of the previous theorem is to deal with the various sources of dependence. The dependence of the noise variables $\varepsilon_i = Y_i - f_0(\mathbf{X}_i)$ on $\mathbf{X}_i$ does not cause major issues, see the proof of Lemma 7.2 for details. However, evaluating the kernel density estimator at two deterministic points $\widehat{f}_{\mathrm{KDE}}(\mathbf{x})$ and $\widehat{f}_{\mathrm{KDE}}(\mathbf{x}')$ leads to highly dependent random variables if $\mathbf{x}$ and $\mathbf{x}'$ are close and the dependence does not vanish even if $\mathbf{x}$ and $\mathbf{x}'$ are far away. The rationale behind the latter is that if $\widehat{f}_{\mathrm{KDE}}(\mathbf{x}) > f_0(\mathbf{x})$, then it is a bit more likely that $\widehat{f}_{\mathrm{KDE}}(\mathbf{x}') < f_0(\mathbf{x}')$ as $\int \widehat{f}_{\mathrm{KDE}}(\mathbf{x}) - f_0(\mathbf{x})\, d\mathbf{x} = 1 - 1 = 0$. To control this dependence, it is common to use Poissonization techniques, cf. [72] Section 3.5.2., [26], and [23] Section 8.3. To explain the idea underlying Poissonization, consider a Poisson point process on $[0,1]^d$. For two disjoint sets $A, B \subset [0,1]^d$, the number of points that fall in set A and the number of points that fall in set B are independent random variables. Also many statistics can be shown to produce independent random variables if they are separately applied to the points in A and the points in B. The Poissonization trick is now to pretend that we do not have n data points $\mathbf{X}'_1, \ldots, \mathbf{X}'_n$ but $\mathcal{M}$ data points, that is, we observe $\mathbf{X}'_1, \ldots, \mathbf{X}'_{\mathcal{M}}$ for $\mathcal{M}$ an independently generated Poisson random variable with intensity n. Then, $\mathbf{X}'_1, \ldots, \mathbf{X}'_{\mathcal{M}}$ can be interpreted as the points of a Poisson point process with intensity $\mathbf{x} \mapsto nf_0(\mathbf{x})$. We can now also redefine the kernel density estimator $\widehat{f}_{\mathrm{KDE}}$ for $\mathbf{X}'_1, \ldots, \mathbf{X}'_{\mathcal{M}}$, by keeping the same normalization $1/n$ but summing over $\ell = 1, \ldots, \mathcal{M}$. Because we have chosen the kernel to have support in $[-1,1]$, $\widehat{f}_{\mathrm{KDE}}(\mathbf{x})$ only depends on the

subset $D(\mathbf{x}) := \{\mathbf{X}'_i : |\mathbf{X}'_i - \mathbf{x}|_\infty \leq h_n, i = 1, \ldots, \mathcal{M}\} \subseteq \{\mathbf{X}'_1, \ldots, \mathbf{X}'_{\mathcal{M}}\}$. If $|\mathbf{x} - \mathbf{y}|_\infty > 2h_n$, then $D(\mathbf{x})$ and $D(\mathbf{y})$ are disjoint sets and one can even show that the statistics $\widehat{f}_{\mathrm{KDE}}(\mathbf{x})$ and $\widehat{f}_{\mathrm{KDE}}(\mathbf{y})$ are independent. To control the change of probability going from n to $\mathcal{M}$ observations, we can apply the following result:

Lemma 3.2. *For $\mathcal{M}$ and $\mathbf{X}'_1, \mathbf{X}'_2, \ldots$ as above, for any function h, and any measurable set A,*

$$\mathbb{P}\Big(\sum_{i=1}^{n} h(\mathbf{X}'_i) \in A\Big) \leq \sqrt{2e\pi n}\,\mathbb{P}\Big(\sum_{i=1}^{\mathcal{M}} h(\mathbf{X}'_i) \in A\Big).$$

Proof of Lemma 3.2. We have

$$\mathbb{P}\Big(\sum_{i=1}^{n} h(\mathbf{X}'_i) \in A\Big) = \mathbb{P}\Big(\sum_{i=1}^{\mathcal{M}} h(\mathbf{X}'_i) \in A \,\Big|\, \mathcal{M} = n\Big) \leq \frac{\mathbb{P}\big(\sum_{i=1}^{\mathcal{M}} h(\mathbf{X}'_i) \in A\big)}{\mathbb{P}(\mathcal{M} = n)}.$$

Since $\mathcal{M}$ is a Poisson(n) random variable we have that $\mathbb{P}(\mathcal{M} = n) = n^n e^{-n}/n!$. By Stirling's formula, see for example [60], $n! \leq \sqrt{2\pi n}(n/e)^n e^{1/(12n)} \leq \sqrt{2e\pi n}(n/e)^n$ and $1/\mathbb{P}(\mathcal{M} = n) \leq \sqrt{2e\pi n}$. □

While Poissonization removes dependence, the factor $\sqrt{2e\pi n}$ arises in the bounds.

3.1. Neural networks

We study the effect of fitting a deep ReLU network in the regression step of the proposed two-step procedure. We rely on the mathematical formulation of deep neural networks introduced in [61] and briefly recall the details for completeness of the exposition. The rectified linear unit (ReLU) activation function is $\sigma(x) := \max\{x, 0\}$. For any vectors $\mathbf{v} = (v_1, \ldots, v_r)^\top, \mathbf{y} = (y_1, \ldots, y_r)^\top \in \mathbb{R}^r$, we define the shifted activation function $\sigma_{\mathbf{v}}\mathbf{y} := (\sigma(y_1 - v_1), \ldots, \sigma(y_r - v_r))^\top$. The number of hidden layers is specified by L and the width of the layers is denoted by the width vector $\mathbf{p} = (p_0, \ldots, p_{L+1}) \in \mathbb{N}^{L+2}$. A network with network architecture $(L, \mathbf{p})$ is any function of the form

$$\mathbf{f} : \mathbb{R}^{p_0} \to \mathbb{R}^{p_{L+1}},\ \mathbf{x} \mapsto \mathbf{f}(\mathbf{x}) = W_L \sigma_{\mathbf{v}_L} W_{L-1} \sigma_{\mathbf{v}_{L-1}} \ldots W_1 \sigma_{\mathbf{v}_1} W_0 \mathbf{x}, \tag{3.2}$$

where W_j is a $p_{j+1} \times p_j$ weight matrix and $\mathbf{v}_j \in \mathbb{R}^{p_j}$ is a shift vector. We use the convention that $\mathbf{v}_0 := (0, \ldots, 0)^\top \in \mathbb{R}^{p_0}$. Denote the maximum entry norm of a matrix W by $\|W\|_\infty$. The class of ReLU networks with architecture $(L, \mathbf{p})$ and parameters bounded in absolute value by one is

$$\mathcal{F}(L, \mathbf{p}) := \Big\{\mathbf{f} \text{ is of the form (3.2)} \ : \max_{j \in \{0, \ldots, L\}} \big(\|W_j\|_\infty \vee |\mathbf{v}_j|_\infty\big) \leq 1\Big\}.$$

For a matrix W denote the counting norm (number of non-zero entries) of W by $\|W\|_0$. We are interested in sparsely connected neural networks where the

number of non-zero or active parameters is small compared to the total number of parameters. For this we define the class of s-sparse networks, that are bounded in uniform norm by F, as

$$\mathcal{F}(L,\mathbf{p},s,F) := \Big\{ \mathbf{f} \text{ is of the form (3.2)} \ : \max_{j\in\{0,\dots,L\}} \big(\|W_j\|_\infty \vee |\mathbf{v}_j|_\infty\big) \le 1, \sum_{j=0}^{L} \|W_j\|_0 + |\mathbf{v}_j|_0 \le s, \||\mathbf{f}|_\infty\|_\infty \le F \Big\}.$$

Definition 3.3 (Two-stage neural network density estimator)**.** If the two-stage nonparametric density estimator $\widehat{f}$ fits in the second step a neural network from the class $\mathcal{F}(L,\mathbf{p},s,F)$ to the augmented sample $(\mathbf{X}_1,Y_1),\dots,(\mathbf{X}_n,Y_n)$, then we refer to $\widehat{f}$ as two-stage neural network density estimator. If the kernel in the first step of the procedure is of order s, we call $\widehat{f}$ a two-stage neural network density estimator with kernel of order s.

The larger the sample size n, the more parameters we can fit. The convergence guarantees below suggest choices for the quantities $L,\mathbf{p},s$ that increase in n and depend on structural properties of the true density f.

3.2. Structural constraints: compositions of functions

Deep neural networks are built by composing individual layers. Previously derived statistical theory has shown that they are well-suited to pick up compositional structure in the regression function, [39, 57, 7, 61, 40]. In this work we follow the composition structure introduced in [61] and impose it on the multivariate density f_0, that is, we assume that $f_0 = g_q \circ g_{q-1} \circ \dots \circ g_1 \circ g_0$, with $g_i : [a_i,b_i]^{d_i} \to [a_{i+1},b_{i+1}]^{d_{i+1}}$. Denote by $g_i = (g_{ij})_{j=1,\dots,d_{i+1}}^\top$ the components of g_i and let t_i be the maximal number of variables on which each of the g_{ij} depends. It always holds that $t_i \le d_i$ and for certain models, t_i can be much smaller than d_i. Section 4 provides examples of densities where this is the case. As we consider density estimation on $[0,1]^d$, it follows that $d_0 = d$, $a_0 = 0$, $b_0 = 1$ and $d_{q+1} = 1$. Since g_{ij} depends on t_i variables, we also interpret g_{ij} as a function $[a_i,b_i]^{t_i} \to [a_{i+1},b_{i+1}]$ whenever this is convenient. Denote by α_i the smoothness of each of the functions g_{ij}. Then $g_{ij} \in \mathcal{H}_{t_i}^{\alpha_i}([a_i,b_i]^{t_i},Q_i)$ and the space of compositions of these smooth functions is given by

$$\mathcal{G}(q,\mathbf{d},\mathbf{t},\boldsymbol{\alpha},Q') := \Big\{ f = g_q \circ \dots \circ g_0 : g_i = (g_{ij})_j : [a_i,b_i]^{d_i} \to [a_{i+1},b_{i+1}]^{d_{i+1}}, \ g_{ij} \in \mathcal{H}_{t_i}^{\alpha_i}([a_i,b_i]^{t_i},Q'), \text{ for some } |a_i|,|b_i| \le Q' \Big\}, \tag{3.3}$$

with $\mathbf{d} := (d_0,\dots,d_{q+1})$, $\mathbf{t} := (t_0,\dots,t_q)$, and $\boldsymbol{\alpha} := (\alpha_0,\dots,\alpha_q)$.

If two functions $h,g : \mathbb{R} \to \mathbb{R}$ have respective smoothness $\alpha_h,\alpha_g \le 1$ then it follows from the definition of the Hölder space that the composition $f := g \circ h$ has smoothness at least $\alpha_h\alpha_g$. For $\alpha_h > 1$ or $\alpha_f > 1$, this is not necessarily true

anymore. It turns out that the convergence rates for a compositional function in $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\alpha}, Q')$ are governed by a notion of effective smoothness indices which are defined as

$$\alpha_i^* := \alpha_i \prod_{\ell=i+1}^{q} (\alpha_\ell \wedge 1).$$

Indeed, in the nonparametric regression model with i.i.d. observations the minimax estimation rate is up to $\log(n)$-terms

$$\phi_n := \max_{i=0,\ldots,q} n^{-\frac{2\alpha_i^*}{2\alpha_i^*+t_i}}, \tag{3.4}$$

cf. Theorem 1 and Theorem 3 in [61]. A function can be represented as a composition in different ways. In the function representation $f = g_q \circ \ldots \circ g_0$, the α_i, t_i and the components $g_0, \ldots, g_q$ are not identifiable. Since we are only interested in estimating the density f_0 this does not constitute a problem.

The oracle inequality in Theorem 3.1 together with the approximation and covering entropy bound results for deep ReLU networks from [61] yields a convergence rate result for the proposed two-stage neural networks estimator. Recall that $\Delta_n(\widehat{f}_n, f_0)$ is the optimization error defined in (2.4).

Theorem 3.4 (Convergence rates)**.** *For $n \geq 3$, consider the density estimation model defined by (2.1)-(2.3) with density f_0 in the Hölder class $\mathcal{C}_d^\beta([0,1]^d, Q) \cap \mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\alpha}, Q)$. Let $\widehat{f}_n$ be a two-stage neural network density estimator with kernel of order $\lfloor \beta \rfloor$ as defined in Definition 3.3 for the neural network class $\mathcal{F}(L, (p_0, \ldots, p_{L+1}), s, F)$ with parameters satisfying*

(i) $F \geq \max\{Q, 1\}$,
(ii) $\sum_{i=1}^q \frac{\alpha_i + t_i}{2\alpha_i^* + t_i} \log_2(4t_i \vee 4\alpha_i) \log_2(n) \leq L \lesssim n\phi_n$,
(iii) $n\phi_n \lesssim \min_{i=1,\ldots,L} p_i$,
(iv) $s \asymp n\phi_n \log(n)$.

Then there exists a constant C_4, only depending on $q, \mathbf{d}, \boldsymbol{\alpha}, \mathbf{t}, F, \beta, K$ and the implicit constants in (ii), (iii), and (iv), such that

$$R(\widehat{f}_n, f_0) \leq C_4 L \max\left(\phi_n \log^4(n), n^{-2\beta/d}\right) + 6\Delta_n(\widehat{f}_n, f_0).$$

Any admissible compositional structure $f = g_q \circ \ldots \circ g_0$ leads to an upper bound on the risk. The estimator achieves therefore the fastest convergence rate among all possible representations.

To analyze the estimation risk, we will now ignore the optimization error $\Delta_n(\widehat{f}_n, f_0)$ and focus on the statistical estimation rate $L \max(\phi_n \log^4(n), n^{-2\beta/d})$. Choosing depth $L \asymp \log(n)$, the convergence rate for the learned network $\widehat{f}$ is thus $\phi_n + n^{-2\beta/d}$, up to $\log(n)$-factors. The $n^{-2\beta/d}$-term is due to the kernel density estimator in the first step and already occurs in the general oracle inequality, see also the discussion after Theorem 3.1.

If the density exhibits a compositional structure, it is now of interest to understand which of the two terms ϕ_n and $n^{-2\beta/d}$ will drive the convergence

rate. If the compositional structure is strong enough to make ϕ_n small but β is small compared to d, then $n^{-2\beta/d}$ dominates the convergence rate. This is faster than the standard nonparametric rate $n^{-2\beta/(2\beta+d)}$ for estimation of a β-smooth function but still suffers from the curse of dimensionality.

If $2\beta \geq d$, then $n^{-2\beta/d} = O(n^{-1})$. Since $\phi_n \gg n^{-1}$, the rate is in this case always of order ϕ_n (up to log-factors). The condition $2\beta \geq d$ appears frequently in the literature on nonparametric statistics and empirical risk minimization. For $d = 1$, $2\beta > 1$ is known to be a necessary condition for nonparametric density estimation and nonparametric regression to be asymptotically equivalent if all densities are bounded from below [53, 58]. The condition $2\beta \geq d$ seems also necessary to ensure that the nonparametric least squares estimator achieves the optimal nonparametric rate $n^{-2\beta/(2\beta+d)}$, see e.g. Section 6.1 in [63]. Barron [6] showed that shallow neural networks can circumvent the curse of dimensionality under a Fourier criterion. A sufficient, but not necessary condition for this Fourier criterion to be finite is that the partial derivatives up to the least integer β such that $2\beta \geq d + 2$ are square-integrable, see Example 15 in Section IX of [5].

Instead of the proposed two-step method, it seems tempting to further iterate the estimation procedure by generating new response variables $Y_i' := \widehat{f}_n(\mathbf{X}_i)$, $i = 1, \ldots, n$, from the estimator $\widehat{f}_n$ and running another neural network fit on the newly generated supervised sample $(\mathbf{X}_1, Y_1'), \ldots, (\mathbf{X}_n, Y_n')$. We believe that this can, however, not improve the convergence rate. The reason is that the new network fit cannot decrease the bias that was already present in the estimator $\widehat{f}_n$. The rate in Theorem 3.4 is obtained by balancing different terms. In particular the squared approximation error that is closely related to the squared bias is of the order of the convergence rate. Thus, if the bias cannot be reduced by another neural network fit also the convergence rate cannot be improved.

In the next section, we provide more explicit examples of densities that satisfy the compositional assumption and attain the convergence rate ϕ_n.

4. Examples of multivariate densities with compositional structure

Compositional structures arise naturally in density modelling. One possibility to see this is to rewrite the joint density f as a product

$$f(x_1, \ldots, x_d) = f(x_d|x_1, \ldots, x_{d-1}) \cdot \ldots \cdot f(x_2|x_1) f(x_1).$$

Each factor $f(x_i|x_1, \ldots, x_{i-1})$ is a function of i variables. But the effective number of variables can be much smaller under conditional independence of the variables. When $\mathbf{X} = (X_1, \ldots, X_d)^\top$ is generated for instance from a Markov chain, X_i only depends on X_{i-1} and the density is a product of bivariate conditional densities

$$f(x_1, \ldots, x_d) = f(x_d|x_{d-1}) \cdot \ldots \cdot f(x_2|x_1) f(x_1). \tag{4.1}$$

Such a structure could occur if the individual data vectors are recordings from a time series, that is, every observation $\mathbf{X}_i = (X_{i,1}, \ldots, X_{i,d})^\top$ contains measurements of the same quantity taken at d different time instances. We now assume

that the density is of the form

$$f(x_1,\dots,x_d)=\prod_{I\in\mathcal{R}}\psi_I(x_I), \tag{4.2}$$

with $\mathcal{R}\subseteq\{S\subset\{1,\dots,d\}, 1\le|S|\le r\}$, r a given number, $x_I=(x_i)_{i\in I}$, and ψ_I non-negative functions. Observe that $|\mathcal{R}|\le\sum_{s=1}^{r}\binom{d}{s}$.

Lemma 4.1. *Consider a density f of the form (4.2). If all the functions ψ_I in the decomposition satisfy $\psi_I\in\mathcal{H}^{\gamma}_{r_I}([0,1]^{r_I},Q)$ for some $r_I\le r$, then the density f can be rewritten as a composition $g_1\circ g_0$ of the form (3.3), with $(d_0,d_1)=(d,|\mathcal{R}|)$, $(t_0,t_1)=(r,|\mathcal{R}|)$, $(\alpha_0,\alpha_1)=(\gamma,\zeta)$, and ζ arbitrarily large.*

Ignoring here and in the rest of this section the optimization error, under the combined conditions of Lemma 4.1 and Theorem 3.4, the proposed two-stage neural network density estimator achieves, up to $\log(n)$-factors, the convergence rate

$$n^{-\frac{2\gamma}{2\gamma+r}}\vee n^{-\frac{2\beta}{d}}, \tag{4.3}$$

with β the (global) Hölder smoothness of the joint density f. If $\beta=\gamma$, that is, the effective smoothness γ coincides with the global Hölder smoothness β of f, then the achieved rate is $n^{-\frac{2\gamma}{2\gamma+r}}$ if $\gamma\ge(d-r)/2$ and $n^{-\frac{2\beta}{d}}$ if $\gamma\le(d-r)/2$. We always have $\beta\ge\gamma$. If $\beta>\gamma$, we conjecture that in most cases there exists a different factorization of the density f with β-smooth ψ_I.

Next, we discuss three examples of models that are of the form (4.2).

Independent variables: If $\mathbf{X}=(X_1,\dots,X_d)$ is a vector containing independent random variables, the joint density is given by

$$f(x_1,\dots,x_d)=\prod_{i=1}^{d}f_i(x_i), \tag{4.4}$$

where f_i is the marginal density of X_i. We assume that f_i is α-Hölder smooth. If we are unaware of the independence and simply use multivariate kernel density estimators to estimate f, we will suffer from the curse of dimensionality as demonstrated for Gaussian densities and Gaussian kernels in Chapter 7 of [65].

Observe that (4.4) is of the form (4.2), with $\mathcal{R}$ the set of singletons. Thus under the combined conditions of Lemma 4.1 and Theorem 3.4, we get, up to $\log(n)$-factors, the convergence rate $n^{-2\alpha/(2\alpha+1)}\vee n^{-2\beta/d}$, with β the (global) Hölder smoothness of the joint density f. The construction in Lemma 4.1 implies that $\beta\ge\alpha$. The next result shows that in this case we necessarily have equality $\beta=\alpha$. In other words the smoothness of the joint density f has to be equal to the (effective) smoothness of the least smooth marginal density.

Lemma 4.2. *Let $\alpha>0$. Consider a density f of the form (4.4) with f_i, $i=1,\dots,d$ probability density functions on $[0,1]$. If f is α-Hölder smooth, then $f_1,\dots,f_d$ are α-Hölder smooth.*

Graphical models: Let $(X_1, \ldots, X_d)$ be a d-dimensional random vector. An undirected graphical model (or Markov random field) is defined by a graph with d nodes representing the d random variables. In this graph, no edge between node i and j is drawn if and only if X_i, X_j are conditionally independent given all the other variables $\{X_1, \ldots, X_d\} \setminus \{X_i, X_j\}$. A clique in a graph is any fully connected subgraph. When the joint density $f(x_1, \ldots, x_d)$ is strictly positive with respect to a σ-finite product measure, the Hammersley-Clifford theorem states that

$$f(x_1, \ldots, x_d) = \prod_{C \in \mathcal{C}} \psi_C(x_C), \tag{4.5}$$

where $\mathcal{C}$ is the set of all cliques in the graph and ψ_C are suitable functions called potentials [10, 44]. As we consider densities supported on $[0,1]^d$, one can take as dominating product measure the uniform distribution on $(0,1)^d$ and the condition requires that the density is strictly positive on $(0,1)^d$. There is no clear link between the potentials and marginal densities.

Assuming that the true density f_0 satisfies (4.5) with largest clique size r and all potentials having Hölder smoothness γ, Lemma 4.1 implies that, under the conditions of Theorem 3.4, the two-stage neural network density estimator is able to exploit the underlying low-dimensional structure and achieves the rate $n^{-2\gamma/(2\gamma+r)} \vee n^{-2\beta/d}$, up to $\log(n)$-factors.

Bayesian networks: Bayesian network models are widely used to model for instance medical expert systems [41, 32] and causal relationships [56]. As in the previous section, consider a d-dimensional random vector $(X_1, \ldots, X_d)$. In a Bayesian network, the dependence relationships of the variables are encoded in a directed acyclic graph with nodes $\{1, \ldots, d\}$ [56, 41, 11, 42]. A directed acyclic graph (DAG) is a directed graph that contains no cycles, meaning one cannot visit the same node twice by following a path along the direction of the edges. The parents $\mathrm{pa}(i)$ of a node i are all nodes that have an edge pointing to node i. The ancestors of node i are all nodes j such that there exists a path along the direction of edges that starts at node j and ends at node i.

The DAG underlying a Bayesian network is constructed such that each variable X_i is conditionally independent of all its ancestors given the parents $X_{\mathrm{pa}(i)} := \{X_j : j \in \mathrm{pa}(i)\}$ in the graph. The joint density can now be written as product of conditional densities

$$f(x_1, \ldots, x_d) = f_d\big(x_d | x_{\mathrm{pa}(d)}\big) \cdot \ldots \cdot f_1\big(x_1 | x_{\mathrm{pa}(1)}\big). \tag{4.6}$$

In particular, if $X_1, \ldots, X_d$ are generated from a Markov chain, this can be represented by the DAG $X_1 \to X_2 \to \ldots \to X_d$. Thus $\mathrm{pa}(j) = \{j-1\}$ for $j > 1$, and we obtain the factorization property $f(x_1, \ldots, x_d) = f(x_d|x_{d-1}) \cdot \ldots \cdot f(x_2|x_1) f(x_1)$.

Assuming that the true density f_0 satisfies (4.6), that no node in the DAG has more than r parents, and all conditional densities $f_d\big(x_i | x_{\mathrm{pa}(i)}\big)$ have Hölder smoothness γ, Lemma 4.1 shows that, under the conditions of Theorem 3.4, the two-stage neural network estimator achieves the convergence rate $n^{-2\gamma/(2\gamma+r+1)} \vee n^{-2\beta/d}$, up to $\log(n)$-factors.

4.1. Copulas

Copulas are widely employed to model dependencies between variables and to construct multivariate distributions, [52, 18, 19]. Denote by F the multivariate distribution function, with marginals $F_1(x_1), \dots, F_d(x_d)$ and density f. Sklar's theorem states that there exists a (unique) d-dimensional copula C (a multivariate distribution function with uniformly distributed marginals on $[0,1]$) such that $F(\mathbf{x}) = C(F_1(x_1), \dots, F_d(x_d))$. The density f can then be rewritten by the chain rule as

$$f(\mathbf{x}) = c\big(F_1(x_1), \dots, F_d(x_d)\big) \prod_{i=1}^{d} f_i(x_i), \tag{4.7}$$

where $f_i(x_i) = F_i'(x_i)$ is the marginal density with respect to x_i and c is the density of C (assuming that all these densities exist). For a reference, see Section 2.3 of [52].

Lemma 4.3. *Consider a density f of the form (4.7). If $c \in \mathcal{H}_d^{\gamma_c}([0,1]^d, Q_c)$ and $f_i \in \mathcal{H}_1^{\gamma_0}([0,1], Q)$, for $i = 1, \dots, d$, then, the density f can be rewritten as a composition $g_2 \circ g_1 \circ g_0$ of the form (3.3), with $(d_0, d_1, d_2) = (d, 2d, d+1)$, $(t_0, t_1, t_2) = (1, d, d+1)$, $(\alpha_0, \alpha_1, \alpha_2) = (\gamma_0, \gamma_c, \gamma)$, and γ arbitrarily large.*

Assume that the true density is of the form (4.7), that $\beta = \gamma_c \wedge \gamma_0$, and that all the conditions on the kernel and the network architecture underlying Theorem 3.4 are satisfied. Applying the decomposition of the density in Lemma 4.3, Theorem 3.4 yields the convergence rate $n^{-2\gamma_0/(2\gamma_0+1)} \vee n^{-2\gamma_c/(2\gamma_c+d)} \vee n^{-2\beta/d}$, up to $\log(n)$-factors. When γ_c and γ_0 satisfy $\gamma_c/d \geq \gamma_0 \geq (d-1)/2$, the convergence rate becomes $n^{-2\gamma_0/(2\gamma_0+1)}$ (up to $\log(n)$-factors). If instead, the copula density c is smoother than the marginals, in the sense that $\gamma_c > \gamma_0 = \beta$, then the obtained convergence rate is faster than the standard nonparametric rate $n^{-2\beta/(2\beta+d)}$ for estimation of β-Hölder smooth functions.

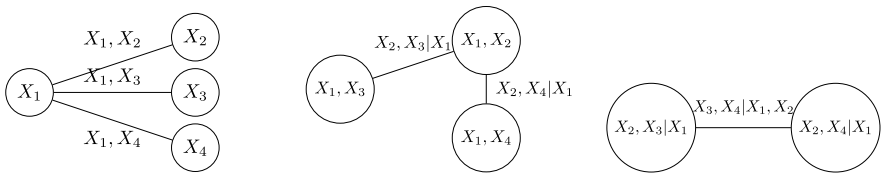
As example, consider the d-variate Farlie-Gumbel-Morgenstern copula family with parameter vector $\boldsymbol{\theta}$, which has copula density

$$c_{\boldsymbol{\theta}}(u_1, \dots, u_d) = 1 + \sum_{r=2}^{d} \sum_{1 \leq j_1 < \cdots < j_r \leq d} \theta_{j_1 \dots j_r} \prod_{k=1}^{r} (1 - 2u_{j_k}),$$

for a parameter vector $\boldsymbol{\theta}$ satisfying $|\boldsymbol{\theta}|_\infty \leq 1$ and

$$1 + \sum_{r=2}^{d} \sum_{1 \leq j_1 < \cdots < j_r \leq d} \theta_{j_1 \dots j_r} \prod_{k=1}^{r} \xi_{j_k} \geq 0 \quad \text{for all } \xi_{j_k} \in \{-1, 1\},$$

[34, 22, 24]. The double summation sums over all $2^d - d - 1$ subsets of $\{1, \dots, d\}$ with at least two elements. Since the input of the copula comes from the distribution functions of the marginals, it holds that $(u_1, \dots, u_d) \in [0,1]^d$. This implies $v_j := (1 - 2u_j) \in [-1, 1]$, and by Lemma 8.1, $\mathbf{v} \mapsto \prod_{k=1}^{r} v_{j_k} \in \mathcal{H}_d^{\gamma_c}([-1,1]^d, 2^d)$, for all $\gamma_c \geq d+1$. Together with the chain rule, this yields $\mathbf{u} \mapsto \prod_{k=1}^{r} (1 - 2u_{j_k}) \in$



(a) First tree (b) Second tree (c) Third tree

Fig 1: Example of a regular vine on four variables. Another example is given in Figure 2.

$\mathcal{H}_d^{\gamma_c}([-1,1]^d, 4^d)$. The derivative of a sum is the sum of the derivatives and therefore the triangle inequality and $|\boldsymbol{\theta}|_\infty \leq 1$ imply for the copula density $c_{\boldsymbol{\theta}} \in \mathcal{H}_d^{\gamma_c}([-1,1]^d, (2^d - d)4^d)$, for all $\gamma_c \geq d+1$. For this family of copulas, the effective smoothness of the composition is thus determined by the smoothness of the marginals and the convergence rate becomes $n^{-2\gamma_0/(2\gamma_0+1)} \vee n^{-2\beta/d}$.

Explicit low-dimensional copula structures can be imposed using the fact that a d-dimensional copula density factorizes into a product of $d(d-1)/2$ bivariate (conditional) copula densities [51, 8, 1, 20]. The key ingredient in this argument is to successively rewrite the conditional densities using the formula $f_{X|Y}(x|y) = c_{X,Y}(F_X(x), F_Y(y))f_X(x)$, where $c_{X,Y}$ denotes the bivariate copula density of (X, Y). The decomposition into bivariate copulas is not unique. Already for three variables (X, Y, Z), there are two possible decompositions, namely

$$f_{X|Y,Z}(x|y,z) = c_{X,Y|Z}\big(F_{X|Z}(x|z), F_{Y|Z}(y|z) \,|\, z\big) f_{X|Z}(x|z)$$

and a second decomposition that interchanges the roles of y and z. The so-called simplifying assumption [70, 51, 20] states that all the bivariate copulas in the decomposition are independent of the conditioned variables, in other words

$$c_{i,j|k}\big(F_{i|k}(x_i|x_k), F_{j|k}(x_j|x_k) \,\big|\, x_k\big) = c_{i,j|k}\big(F_{i|k}(x_i|x_k), F_{j|k}(x_j|x_k)\big).$$

For the remainder of this section, we will assume that the simplifying assumption holds.

A way to define such decompositions is by relying on regular vines, [51, 8, 1, 20]. A vine on d variables $X_1, \ldots, X_d$ is a set of trees $(T_1, \ldots, T_r)$, such that the nodes of the first tree T_1 are $u_1, \ldots, u_d$. The nodes of the tree T_i, for $i = 2, \ldots, r$, are (a subset of) the edges of the tree T_{i-1}. For a regular vine it furthermore holds that $r = d - 1$, that two edges in a tree can only be joined by an edge in the next tree if these edges share a common node, and that the set of nodes of T_i has to be equal to the set of edges of T_{i-1}.

Any regular vine on $(X_1, \ldots, X_d)$ defines a factorization of a d-dimensional copula, by associating a bivariate copula density to each edge in any of the trees. Copulas defined in this way are called vine-copulas.

Figure 1 shows an example of a regular vine with four variables. Regular vines such as the one in Figure 1, where each tree has one node that has an edge to all other nodes in that tree, are known as canonical-vines [1] or C-vines [20]. The

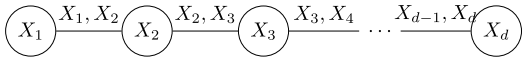


Fig 2: Structure of the first tree of the D-vine copula used in the simulation.

density corresponding to a canonical vine on d variables (up to renumbering the variables) is given by

$$\prod_{k=1}^{d} f_k(x_k) \prod_{j=1}^{d-1} \prod_{i=1}^{d-j} c_{j,j+i|1,\ldots,j-1}\Big(F_{j|1,\ldots,j-1}(x_j|x_1,\ldots,x_{j-1}), F_{j+i|1,\ldots,j-1}(x_{j+i}|x_1,\ldots,x_{j-1})\Big).$$

Another type of regular vine is the D-vine, [1, 20]. In a D-vine no node in any tree is connected to more than two edges. Figure 2 shows the first tree of a D-vine on d variables. The density corresponding to a D-vine on d variables (up to renumbering the variables) is given by

$$\prod_{k=1}^{d} f_k(x_k) \prod_{j=1}^{d-1} \prod_{i=1}^{d-j} c_{i,i+j|i+1,\ldots,i+j-1}\Big(F_{i|i+1,\ldots,i+j-1}(x_i|x_{i+1},\ldots,x_{i+j-1}), F_{i+j|i+1,\ldots,i+j-1}(x_{i+j}|x_{i+1},\ldots,x_{i+j-1})\Big).$$

If two random variables X_1, X_2 are conditionally independent given X_3, then $c_{1,2|3} = 1$. If such conditional independence relations hold, one can simplify the vine-structure. For example consider the vine on four variables in Figure 1. In the (very simplified) case that X_2 and X_3 are independent given X_1, that X_2 and X_4 are independent given X_1, and that X_3 and X_4 are independent given (X_1, X_2), only the bivariate copulas on the edges of the first tree (Figure 1a) appear in the decomposition, cf. Section 3 of [1]. More generally, suppose that there exists a canonical vine on d variables such that the bivariate (conditional) copula densities associated with all the trees except the first one are equal to one, then under the simplifying assumption, the decomposition becomes

$$f(\mathbf{x}) = \prod_{k=1}^{d} f_k(x_k) \prod_{i=2}^{d} c_{1,i}\big(F_1(x_1), F_i(x_i)\big). \tag{4.8}$$

Here X_1 is the root of the first tree, which can always be achieved by renumbering the variables. In the case of a D-vine, the decomposition (up to renumbering) becomes

$$f(\mathbf{x}) = \prod_{k=1}^{d} f_k(x_k) \prod_{i=1}^{d-1} c_{i,i+1}\big(F_i(x_i), F_{i+1}(x_{i+1})\big). \tag{4.9}$$

Lemma 4.4. *Consider a density f of the form (4.8) or (4.9). If $f_i \in \mathcal{H}_1^{\gamma}([0,1], Q)$, for all $i = 1, \ldots, d$, and all bivariate copula densities are in $\mathcal{H}_2^{\gamma_c}([0,1]^2, Q)$,*

then, the function f can be written as a composition $g_2 \circ g_1 \circ g_0$, with $(d_0, d_1, d_2) = (d, 2d, 2d-1)$, $(t_0, t_1, t_2) = (1, 2, 2d-1)$, $(\alpha_0, \alpha_1, \alpha_2) = (\gamma, \gamma_c, \zeta)$, where ζ is arbitrarily large.

If we assume that $\gamma_c = \gamma = \beta$, then under the combined conditions of Theorem 3.4 and Lemma 4.4, the proposed two-stage neural network estimator achieves the convergence rate $n^{-2\gamma/(2\gamma+2)} \vee n^{-2\gamma/d}$ up to $\log(n)$ factors. If $d > 2$, this rate is faster than the nonparametric estimation rate $n^{-2\gamma/(2\gamma+d)}$. Furthermore when $\gamma = \gamma_c \geq d/2-1$, the rate equals $n^{-2\gamma/(2\gamma+2)}$, up to $\log(n)$-factors. If instead we assume that $\gamma_c \geq 2\gamma = 2\beta$, that is, the copulas have at least twice the Hölder smoothness of the marginals, then the rate becomes $n^{-2\gamma/(2\gamma+1)} \vee n^{-2\gamma/d}$, up to $\log(n)$-factors.

4.2. Mixture distributions

If the true density is a mixture and all mixture components can be estimated by a fast convergence rate, it should be possible to also estimate the true density with a fast rate. An example are multi-view models [66, 4, 36, 73], that assume a true density of the form

$$f_0(\mathbf{x}) = \sum_{j=1}^{r} a_j \prod_{k=1}^{d} f_{j,k}(x_k),$$

with non-negative mixture weights $a_1, \dots a_r$ summing up to one and univariate densities $f_{j,k}$, $j = 1, \dots, r$; $k = 1, \dots, d$.

Below we assume more generally that the true density is a mixture density of the form

$$f_0 = a_1 f_1 + \ldots + a_r f_r \tag{4.10}$$

with densities f_j in the compositional Hölder space $\mathcal{G}(q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j, Q')$ defined in (3.3). In particular, we allow the parameters q_j, $\mathbf{d}_j = (d_{0,j}, \dots, d_{q_j+1,j})$, $\mathbf{t}_j = (t_{0,j}, \dots, t_{q_j,j})$, and $\boldsymbol{\alpha}_j = (\alpha_{0,j}, \dots, \alpha_{q_j,j})$ to depend on j. Compositional spaces are not closed under linear combinations and therefore there is no natural embedding of f into the compositional spaces of the f_j's. As shown next, the convergence rate for estimation of f still coincides with the maximum among all convergence rates for estimation of individual mixture components f_j. Set $\alpha^*_{i,j} := \alpha_{i,j} \prod_{\ell=i+1}^{q_j} (\alpha_{\ell,j} \wedge 1)$ and $\phi^\star_n := \max_{j=1,\dots,r} \phi_{n,j}$, where

$$\phi_{n,j} := \max_{i=0,\dots,q_j} n^{-\frac{2\alpha^*_{i,j}}{2\alpha^*_{i,j}+t_{i,j}}}$$

is the rate (3.4) for estimation of f_j.

Theorem 4.5 (Convergence rates for mixture distributions)**.** *Consider the density estimation model defined by (2.1)-(2.3) with density $f_0 = \sum_{i=1}^{r} a_i f_i$, where $a_1, \dots a_r$ are non-negative mixture weights summing up to one, and with*

$f_j \in \mathcal{C}_d^\beta([0,1]^d, Q) \cap \mathcal{G}(q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j, Q)$, *for all* $j = 1, \ldots, r$. *Let* $\widehat{f}_n$ *be a two-stage neural network density estimator with kernel of order* $\lfloor \beta \rfloor$ *as defined in Definition 3.3 for the neural network class* $\mathcal{F}(L, (p_0, \ldots, p_{L+1}), s, F)$ *with parameters satisfying*

(i) $F \geq \max\{Q, 1\}$,
(ii) $\max_{j=1,\ldots,r} \sum_{i=1}^{q_j} \frac{\alpha_{i,j}+t_{i,j}}{2\alpha_{i,j}^*+t_{i,j}} \log_2(4t_{i,j} \vee 4\alpha_{i,j}) \log_2(n) \leq L \lesssim n\phi_n^\star$,
(iii) $n\phi_n^\star \lesssim \min_{i=1,\ldots,L} p_i$,
(iv) $s \asymp n\phi_n^\star \log(n)$.

If n *is large enough, then there exists a constant* C_6, *only depending on* $r, (q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j)_{j=1}^r$, F, β, K *and the implicit constants in (ii), (iii), and (iv) such that*

$$R(\widehat{f}_n, f_0) \leq C_6 L \max\left(\phi_n^\star \log^4(n), n^{-\frac{2\beta}{d}}\right) + 6\Delta_n(\widehat{f}_n, f_0).$$

5. Simulations

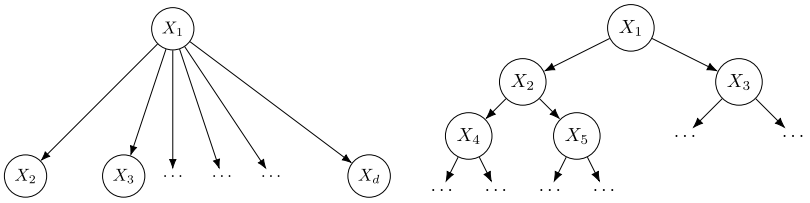
5.1. Methods

In a numerical simulation study we compare the proposed two-stage neural network density estimator (named SD for Split Data) as described in Definition 3.3 to two other methods. The FD (full data) method follows the same construction as the two-stage neural network estimator but uses for both steps the full dataset without sample splitting. Thus, we have twice as many data for the individual steps, but also incur additional dependence between the regression variables as each of the constructed response variables Y_i depends on the entire dataset (instead of only on the kernel dataset and the corresponding X_i from the regression set). The neural network based methods are moreover compared to a multivariate kernel density estimator (KDE).

As suggested by the theory, for the first step in the SD and FD method, the bandwidths for the kernel density estimator are chosen of the form $c_1(\log(n)/n)^{1/d}$ and $c_2(\log(2n)/(2n))^{1/d}$. For the KDE method, the bandwidth is $c_3 n^{-1/(2\beta+d)}$. The constants c_1, c_2, c_3 are determined based on the average of the optimal bandwidths found by 50-fold cross-validation, taking as search space the interval $[0.05, 1.1]$ with stepsize 0.005, on five independently generated datasets with sample size $n = 200$ from the true density. Taking $n = 200$ for the calibration is natural as it is the smallest sample size in the simulation environment.

5.2. Densities

For the different simulation settings, we generate data from five densities. These densities are called Naive Bayes mixing (NBm), Naive Bayes shifting (NBs), Binary Tree mixing (BTm), Binary Tree shifting (BTs) and Copula (C).



(a) Naive Bayes DAG (b) Binary tree DAG

Fig 3: DAG for the Naive Bayes network (a) and the Bayesian network with binary tree structure (b).

5.2.1. NBm, NBs, BTm, BTs

The densities (NBm) and (NBs) are so-called Naive Bayes networks [41] with DAGs displayed in Figure 3a and density factorization

$$f(x_1,\dots,x_d) = f_d(x_d|x_1)\cdot\ldots\cdot f_2(x_2|x_1)f_1(x_1). \tag{5.1}$$

The densities (BTm) and (BTs) are Bayesian networks with DAGs displayed in Figure 3b and density factorization

$$f(x_1,\dots,x_d) = f_1(x_1)\prod_{j=2}^{d} f_j(x_j|x_{\lceil (j-1)/2\rceil}). \tag{5.2}$$

For the density f_1, we use the exponential of a standard Brownian motion on $[0,1]$, normalized such that f_1 integrates to one. We use two different types of conditional densities. The mixing conditional density has mixture weights from the conditioned variable,

$$f_j(x_j|x_i) = x_i h_j(x_j) + (1-x_i)h_j(1-x_j), \tag{5.3}$$

with h_j a density supported on $[0,1]$. The shifting conditional density incorporates a shift determined by the conditioned variable,

$$f_j(x_j|x_i) = h_j\big(\max\{x_j - x_i/4, 0\}\big), \tag{5.4}$$

with h_j a density supported on the interval $[0,3/4]$, so that the support of $f_j(\cdot|x_i)$ is ensured to lie in $[0,1]$.

For the densities (NBm) and (BTm) all conditional densities $f_j(\cdot|\cdot)$ in the factorization are mixing densities (5.3). For the densities (NBs) and (BTs) the conditional densities $f_j(\cdot|\cdot)$ in the factorization are shifting densities (5.4) if j is divisible by 3 and mixing densities (5.3) otherwise.

It remains to choose the density h_j in (5.3) and (5.4). We consider scenarios containing both smooth and rough densities. For (NBm), (NBs), (BTm) and (BTs) and all j such that $j-1$ is not divisible by 3, we set

$$h_j(x) = \Big(1 - \frac{2x-1}{d}\Big)\mathbf{1}\big(0 \le x \le 1\big). \tag{5.5}$$

Viewed as functions on $[0,1]$, these densities have arbitrarily large Hölder smoothness. The densities take values between $1-1/d$ and $1+1/d$ ensuring that in higher dimensions the joint densities, which are products, neither become extremely small or large.

For (NBm) and (BTm) and all $j>1$ such that $j-1$ is divisible by 3, we take as densities h_j the exponential of the Brownian motion on $[0,1]$, normalized such that h_j integrates to one. Brownian motion has Hölder smoothness $1/2-\eta$ for any $\eta\in(0,1/2)$, but is almost surely not 1/2-Hölder smooth [48]. This means that these densities have low regularity.

For (NBs) and (BTs) and all $j>1$ such that $j-1$ is divisible by 3, we take as densities h_j the paths of the exponential of the Brownian motion on $[0,1]$ multiplied with the function $x\mapsto\rho(x)=\max(0,(4x/3)(1-4x/3))$ and normalized such that h_j integrates to one. Multiplication with ρ ensures that the support of these densities is in $[0,3/4]$, as required in the definition (5.4).

The conditional densities f_j defined in (5.3) and (5.4) can be interpreted as compositional functions.

Lemma 5.1. *Consider the mixing conditional density f_j in (5.3). If $h_j\in\mathcal{H}_1^{\gamma_j}([0,1],Q)$, then f_j can be written as the composition $g_1\circ g_0$, with $(d_0,d_1)=(2,3)$, $(t_0,t_1)=(1,3)$, and $(\alpha_0,\alpha_1)=(\gamma_j,\zeta)$, with ζ arbitrarily large.*

Lemma 5.2. *Consider the shifting conditional density f_j in (5.4) If $h_j\in\mathcal{H}_1^{\gamma_j}([0,3/4],Q)$, then f_j can be written as $g_1\circ g_0$, with $(d_0,d_1)=(2,1)$, $(t_0,t_1)=(2,1)$, $(\alpha_0,\alpha_1)=(1,\gamma_j)$.*

The (NBm), (NBs), (BTm) and (BTs) joint densities are thus compositions where the components with low regularity are all univariate functions, making the rate ϕ_n dimensionless. The factorization in (5.1) and the composition of Lemma 4.1 combined with the composition in Lemma 5.1 shows this for the (NBm) model. The factorization in (5.1) and the composition Lemma 4.1 combined with the compositions in Lemma 5.1 and Lemma 5.2 show this for the (NBs) model. The factorization in (5.2) and the composition of Lemma 4.1 combined with Lemma 5.1 shows this for the (BTm) model and the factorization in (5.2) and the composition of Lemma 4.1 combined with the compositions in Lemma 5.1 and Lemma 5.2 show this for the (BTs) model.

5.2.2. Simulation setup for copula density model

For the copula model, the density (C) is associated to a D-vine copula of the form (4.9), that is,

$$f(\mathbf{x})=\prod_{k=1}^{d}f_k(x_k)\prod_{i=1}^{d-1}c_{i,i+1}\big(F_i(x_i),F_{i+1}(x_{i+1})\big). \tag{5.6}$$

The bivariate copula densities $c_{i,i+1}$ are chosen from the bivariate Farlie-Gumbel-Morgenstern copula family

$$c_{i,i+1}\big(F_i(x_i),F_{i+1}(x_{i+1})\big)=1+\theta_i\big(1-2F_i(x_i)\big)\big(1-2F_{i+1}(x_{i+1})\big),$$

with parameter $\theta_i := -1 + 2(i-1)/(d-2)$, if $-1 + 2(i-1)/(d-2) \neq 0$ and $\theta_i := 1/100$ otherwise. As shown in Section 4.1, these copula densities have arbitrarily large Hölder smoothness. The marginal densities f_k are displayed in Figure 4. The smoothness of this density is determined by the square root, which

$$f_k(x) = \begin{cases} 1 + \frac{1}{2d} - \frac{1}{d}\sqrt{\frac{1}{4} - x}, & \text{if } 0 \le x < \frac{1}{4} \\ 1 + \frac{1}{2d} - \frac{1}{d}\sqrt{x - \frac{1}{4}}, & \text{if } \frac{1}{4} \le x < \frac{1}{2} \\ 1 - \frac{1}{2d} + \frac{1}{d}\sqrt{\frac{3}{4} - x}, & \text{if } \frac{1}{2} \le x < \frac{3}{4} \\ 1 - \frac{1}{2d} + \frac{1}{d}\sqrt{x - \frac{3}{4}}, & \text{if } \frac{3}{4} \le x \le 1. \end{cases}$$

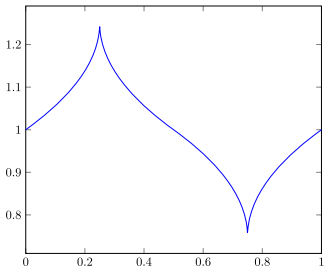
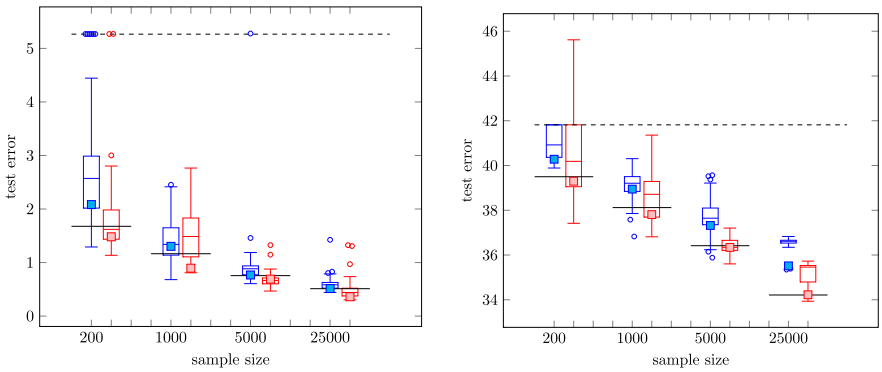


Fig 4: Marginal density $f_k(x)$ in the simulated copula model. The right panel shows the graph of the density for $d = 2$.

has Hölder smoothness 1/2. The right panel of Figure 4 displays the graph for $d = 2$. This marginal density is appealing as it has a closed-form expression for the density and the c.d.f. The dependency on d of the marginals ensures that the marginal densities remain between $1 - 1/d$ and $1 + 1/d$ in order to prevent numerical instability. Since the Farlie-Gumbel-Morgenstern copula is infinitely smooth, we get from Lemma 4.4 that the effective smoothness of the joint density generated from this vine-copula approach is equal to 1/2 and thus the rate ϕ_n in Theorem 3.4 becomes $n^{-1/2}$, up to $\log(n)$-factors.

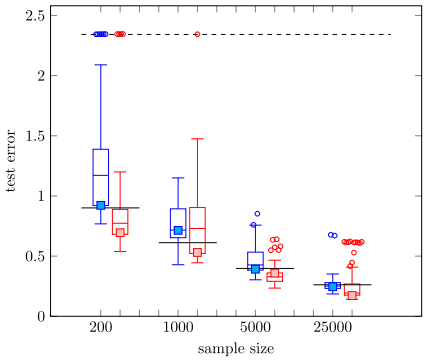
5.3. Neural network training setup

For both the SD and FD method, we train neural networks with width vector $\mathbf{p} = (d, \lceil (2n)^{1/2} \rceil, \lceil (2n)^{1/2} \rceil, \ldots, \lceil (2n)^{1/2} \rceil, 1)$ and depth $L = \lceil \log_2(2n) \rceil$. Since the derived convergence rate of the two-stage neural network estimator is $\phi_n = n^{\eta' - 1/2}$, for any $\eta' \in (0, 1/2)$, in the (NBm), (NBs), (BTm) and (BTs) settings, and $\phi_n = n^{-1/2}$ in the (C) setting, this choice of the network width satisfies the bound in Theorem 3.4. The chosen depth is of the order $\log(n)$ suggested by the theory, but there might be a mismatch regarding the constants in the lower bound of Condition (ii) in Theorem 3.4. Since the proof of this result does not optimize the constants, we find it more appealing to work with the generic choice $L = \lceil \log_2(2n) \rceil$ in the simulations. Furthermore, Theorem 3.4 imposes a sparsity condition on the networks as well as a condition on the maximum norm of the parameters. In the simulation study we use ℓ_2-penalization on the weight matrices and the Glorot uniform initialization [28] to ensure that the parameter values do not become too large. Although these methods do not provide a hard guarantee that the condition on the maximum norm is satisfied, they work reasonably well in practice and the number of learned network parameters exceeding in absolute value one is small compared to the total number

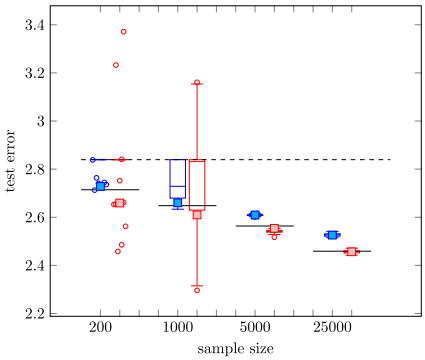


(a) NBs, dimension 4

(b) NBs, dimension 12



(c) NBm, dimension 4



(d) NBm, dimension 12

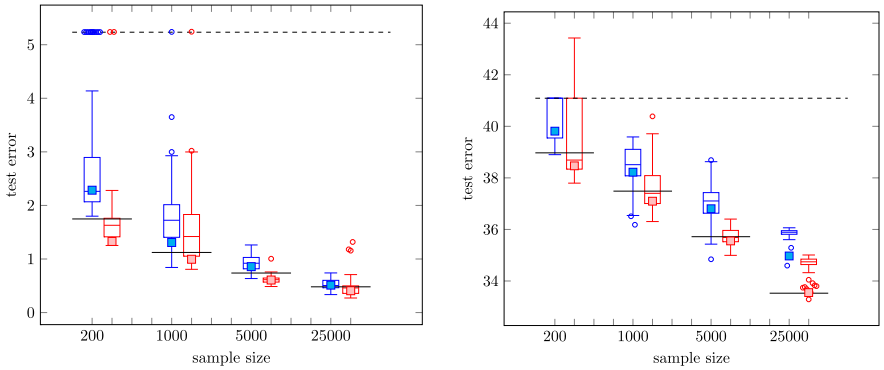
Fig 5: Test errors for the naive Bayes model. SD in blue, FD in red, KDE black bars. The test error of the network with the lowest training error is indicated by the filled square. The black dashed line is the test error of the zero function. Notice that in the individual plots, the y-axis has different starting points.

of network parameters. We use pruning (using the TensorFlow model optimization package) to enforce sparsity. The fraction of zero network parameters is chosen as $1 - 2m\log(m)\phi_m/p$, with p the total number of network parameters and $m = 2n$ for the FD method and $m = n$ for the SD method.

The source code is available on GitHub [12].

5.4. Simulation results

For each of the five densities described in Section 5.2, we generate four training samples, with respective sample sizes $200, 1000, 5000, 25000$. For both the SD and FD method, 50 neural networks are trained with different random initialization on each training sample. Repeating the network fit on the same sample highlights the variation of test performance with respect to the initialization and the achieved training loss. We compare the performance of all the methods



(a) BTs, dimension 4

(b) BTs, dimension 12

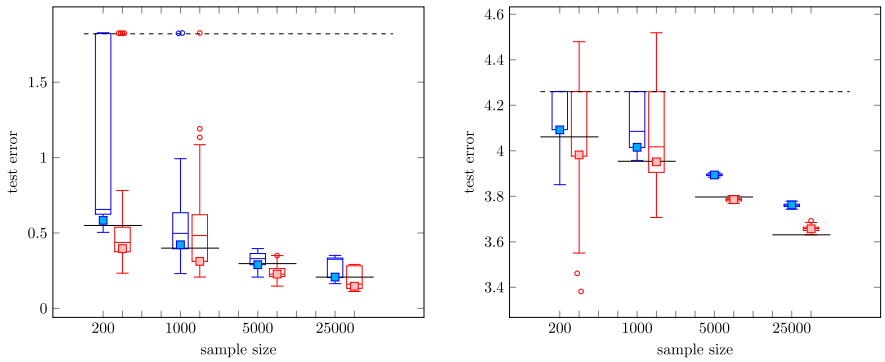
(c) BTm, dimension 4

(d) BTm, dimension 12

Fig 6: Test errors for the Bayesian network model. SD in blue, FD in red, KDE black bars. The test error of the network with the lowest training error is indicated by the filled square. The black dashed line is the test error of the zero function. Notice that in the individual plots, the y-axis has different starting points.

on 10^6 test samples that are always drawn from the same distribution as the training samples. This sample is only used for computing the test error and none of the methods has access to the test samples during training. Figures 5-7 report the test errors for the five different settings.

For the smaller sample sizes, the neural network fit is sometimes the zero function. These reconstructions generate the circles on top of the dashed lines in the plots. The theory claims that among the sparsely connected neural networks that satisfy all the imposed conditions, the one with small training error should perform particularly well. To see whether there is an effect, we mark for every simulation setting the test error of the network with the smallest training error by a filled square. The simulations show that for the FD method, this network fit is often near the first quartile in the box plots and thus indeed performs particularly well among the different random initializations.



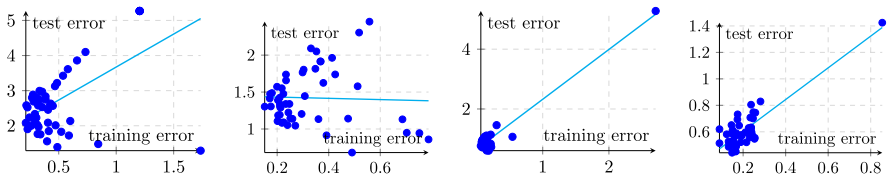
(a) Copula, dimension 4 (b) Copula, dimension 12

Fig 7: Test errors for the Copula model. SD in blue, FD in red, KDE black bars. The test error of the network with the lowest training error is indicated by the filled square. The black dashed line is the test error of the zero function. Notice that in the individual plots, the y-axis has different starting points.

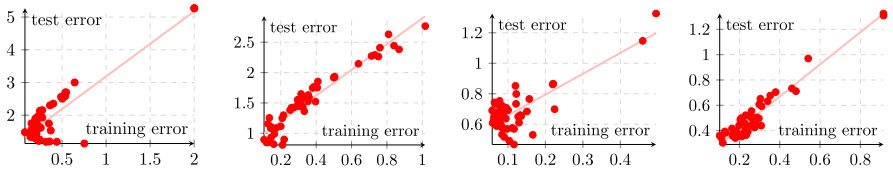
To further investigate the relation between training error and test error, we plot for the (NBs) model in dimension four (Figure 8) and twelve (Figure 9) the training error versus the test error of all networks, for both the SD and FD method and for each of the four considered sample sizes. The linear line displaying the least squares regression fit has positive slope, except for the SD method with sample size 1000 (in both dimensions four and twelve). While fitting a line might not be fully justified given the outliers and the parabola-shaped data, there seems indeed to be a connection between lower training error and improved generalization (lower test error). Interestingly, there are also a few fits with large training error and small test error.

Let us now compare the network fits with the smallest training error (indicated by a blue or red square in Figures 5-7) to the kernel density estimator. To estimate the joint density depending on four variables, the neural network fits based on the FD method with the lowest training error seem to perform best for all sample sizes. For density estimation on $[0,1]^{12}$, the picture is less clear as there are sample sizes for which the KDE method achieves a comparable or even better test error. The test error of the SD method is consistently higher. In dimension 4, it decreases, however, faster than the test errors of the FD and KDE method. Based on the comparison, we do advise to use the two-step method without data splitting and to pick the reconstruction with the smallest training loss based on different random initializations.

While the idea to transform an unsupervised learning problem into a supervised learning problem and using supervised learning methods is appealing, we feel that considerable future effort is required to transform this into stable and efficient algorithms.

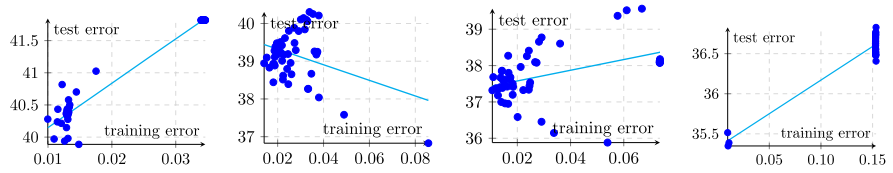


(a) SD method, sample size 200 (b) SD method, sample size 1000 (c) SD method, sample size 5000 (d) SD method, sample size 25000

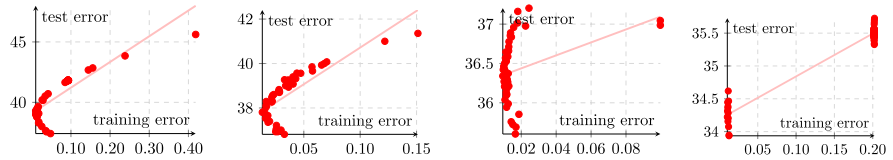


(e) FD method, sample size 200 (f) FD method, sample size 1000 (g) FD method, sample size 5000 (h) FD method, sample size 25000

Fig 8: Scatterplot of the test error versus the training error for the (NBs) model in 4 dimensions. The line shows the linear least squares regression fit.



(a) SD method, sample size 200 (b) SD method, sample size 1000 (c) SD method, sample size 5000 (d) SD method, sample size 25000



(e) FD method, sample size 200 (f) FD method, sample size 1000 (g) FD method, sample size 5000 (h) FD method, sample size 25000

Fig 9: Scatterplot of the test error versus the training error for the (NBs) model in 12 dimensions. The line shows the linear least squares regression fit.

6. Related literature

A more direct method for nonparametric density estimation is to use a class of candidate densities $\mathcal{F}$ and estimate the density by a maximizer of the log-likelihood

$$\arg\max_{f\in\mathcal{F}} \sum_{i=1}^{n} \log f(\mathbf{X}_i),$$

which is equivalent to minimizing the negative log-likelihood or cross-entropy

$$\arg\min_{f\in\mathcal{F}} -\sum_{i=1}^{n} \log f(\mathbf{X}_i).$$

In principle, $\mathcal{F}$ could be a class of neural networks that is normalized or constrained to yield (approximately) probability density functions. For general classes $\mathcal{F}$, nonparametric maximum likelihood estimators have been analyzed in the literature [71]. A major drawback of this approach is the instability in low-density regions that is caused by the divergence of the logarithm $\lim_{x\downarrow 0}\log(x)$. For this reason, all convergence rate results that we are aware of require that the densities are bounded away from zero. This is rather restrictive as for machine learning applications, one often expects large low- or even zero-density regions. Note that the derived convergence guarantees for the proposed two-step nonparametric density estimator do not require the true density to be bounded away from zero.

Methods based on augmenting data via data generated response variables have been explored in various areas in statistics: [37, 43] construct pseudo-outcomes to estimate treatment effects in causal models; [17, 75] use nonparametric methods for imputation of missing data; and [76] deploys pseudo-outcomes for model selection under covariate shift. Relating generative AI to density estimation [54], diffusion models [67, 68] transform the density estimation problem into nonparametric regression of the score functions for different levels of injected noise in the sample.

Our method is inspired by previous work on asymptotic equivalence that links the (unsupervised) nonparametric density estimation problem to a (supervised) regression-type model. More precisely, it is shown that if the univariate densities f are defined on $[0,1]$, are more than $1/2$-smooth and are bounded away from zero, then, the statistical model converges in the Le Cam distance to the statistical problem, where we want to recover f by observing $(Y_t)_{t\in[0,1]}$ with

$$dY_t = 2\sqrt{f(t)}\,dt + n^{-1/2}dW_t, \quad \text{for all } t\in[0,1], \tag{6.1}$$

and W is a Brownian motion. On a high level, convergence in Le Cam distance means that the asymptotic statistical properties are in both models the same. Model (6.1) behaves similarly as observing n i.i.d. pairs (U_i, Y_i) with U_i uniform on $[0,1]$ and $Y_i = 2\sqrt{f(U_i)} + \varepsilon_i$ for independent noise variables $\varepsilon_i \sim \mathcal{N}(0,1)$. This establishes the possibility to transfer nonparametric density estimation into a regression model without losing information regarding asymptotic results.

While the original proof for the asymptotic equivalence statement was non-constructive [53], follow-up work [14, 58] has identified a transformation mapping the observations in the nonparametric density model to the process $(Y_t)_{t\in[0,1]}$ satisfying (6.1). The two key steps in the construction are a Poissonization step, mapping the density estimation problem with n observations to density estimation with $\mathcal{M}\sim\text{Poisson}(n)$ observations, followed by a step that constructs the response variables $(Y_t)_{t\in[0,1]}$ via a Haar wavelet decomposition and a quantile

coupling argument. While the asymptotic equivalence literature motivates our two-step density estimation method and its analysis, there are still many differences as asymptotic equivalence focuses on bounding the Le Cam distance, whereas we are proposing a specific method to use supervised deep learning for nonparametric density estimation.

The proposed two-step procedure is moreover related to Lindsey's method which transforms parametric estimation in exponential families into a Poisson regression problem [45, 46, 25]. The first step of this method discretizes the sample space into disjoint bins. The bin counts follow a multinomial distribution that is then approximated by the Poisson distribution. Assuming Poisson distributed bin counts, maximum likelihood estimation of the parameters results then in a Poisson regression problem. A benefit of Lindsey's transformation is that the normalization constant of the exponential family vanishes. This constant is an integral over the entire domain and hard to compute in high dimensions [49, 27]. While Lindsey's method returns one observation per bin and has been formulated for exponential families, the proposed method in this work focuses on nonparametric densities and artificially creates a supervised dataset by computing a response vector for each of the datapoints. Approximation of the bin counts by the Poisson distribution occurs in our approach in the proof.

While finalizing the article, we became aware of the similar two-step density estimation method [33] proposed in the pattern recognition literature. For the first step, the authors use the band limited maximum likelihood density estimator proposed in [2]. However, this article provides no theory.

Beyond the mentioned connections to asymptotic equivalence, Lindsey's method, and [33], we are unaware of any other density estimation method that is similar to ours. It is important to emphasize that the success of such a two-step procedure relies on a regression method that achieves faster rates than direct density estimation. While this is the case here, for more traditional function spaces, direct density estimation can be shown to be already rate optimal.

7. Proofs for Section 3

Lemma 7.1. *If $n > 1$, then there exists a h_n, such that $(\log(n)/n)^{1/d} \leq h_n \leq 2(\log(n)/n)^{1/d}$ and h_n^{-1} is a positive integer.*

Proof. For all $x \geq 0$, we have $x < 1 + x \leq e^x$ and thus $\log n/n < 1$ as well as $0 < u_n := 2(\log n/n)^{1/d} < 2$ for all $n > 1$. For all $y > 0$, one can find an integer r such that $y/2 \leq 2^r \leq y$. If $y < 2$, we must have $r \leq 0$. Thus, there exists an integer $s \leq 0$ such that $u_n/2 \leq 2^s \leq u_n$. Set $h_n = 2^s$. Since $s \leq 0$, we must have $h_n^{-1} = 2^{-s}$, which is an integer. □

7.1. Proof of Theorem 3.1

The response variables Y_i in the regression model (2.3) are identically distributed, but they are not jointly independent as they all depend through the kernel density estimator on the subsample $(\mathbf{X}'_\ell)_{\ell=1}^n$.

To deal with the dependence induced by the kernel density estimator, we partition the hypercube $[0,1]^d$ into h_n^{-d} hypercubes with sidelength h_n. By construction h_n^{-1} is an integer and therefore no boundary issues arise. The centers of these h_n^{-d} hypercubes are given by the vectors $h_n(k_1-1/2, k_2-1/2, \ldots, k_d-1/2)^\top \in [0,1]^d$ with $k_1, k_2, \ldots, k_d \in \{1, \ldots, h_n^{-1}\}$. By numbering these points (the specific numbering of the points is irrelevant), we assign to each center an index in $\mathcal{J} := \{1, \ldots, h_n^{-d}\}$. The j-th bin $\mathcal{B}_j$ is then the $|\cdot|_\infty$-norm ball of radius $h_n/2$ around the j-th center $C(\mathcal{B}_j)$ in this index set. To avoid that boundary points are in two bins, we include a boundary point only if is not already included in a bin with smaller index in the ordering induced by $\mathcal{J}$. This construction gives a partition of $[0,1]^d$. As each bin is a hypercube with sidelength h_n, the Lebesgue measure is h_n^d (in $\mathbb{R}^d$). The neighborhood of a bin $\mathcal{B}_j$, denoted by $NB(\mathcal{B}_j)$, is the union of all bins whose centers are at most $|\cdot|_\infty$-distance h_n away from the center of $\mathcal{B}_j$, in other words,

$$NB(\mathcal{B}_j) = \bigcup_{\ell : |C(\mathcal{B}_j) - C(\mathcal{B}_\ell)|_\infty \le h_n} \mathcal{B}_\ell \,. \tag{7.1}$$

In two dimensions this neighborhood is also known as the Moore neighborhood. Up to boundary effects, $NB(\mathcal{B}_j)$ is a $|\cdot|_\infty$-ball with radius $\frac{3}{2}h_n$,

$$\begin{aligned}\left\{\mathbf{x} \in [0,1]^d \,:\, |\mathbf{x} - C(\mathcal{B}_j)| < \frac{3}{2}h_n\right\} &\subseteq NB(\mathcal{B}_j) \\ &\subseteq \left\{\mathbf{x} \in [0,1]^d \,:\, |\mathbf{x} - C(\mathcal{B}_j)| \le \frac{3}{2}h_n\right\}.\end{aligned}$$

We further subdivide the bins into equivalence classes. For all sufficiently large n, $h_n \le 1/3$ and the hypercube $[0, 3h_n]^d$ contains exactly 3^d bins. Denote by $(j_s)_{s=1}^{3^d}$ the indices of these bins and define the index set $\mathcal{J}_s \subset \mathcal{J}$ by

$$\mathcal{J}_s := \left\{\ell \in \mathcal{J} : \frac{C(\mathcal{B}_\ell) - C(\mathcal{B}_{j_s})}{3h_n} \in \mathbb{Z}^d\right\}.$$

Suppose there exists $j \in \mathcal{J}_s \cap \mathcal{J}_{s'}$ for $s \neq s'$. Then, it follow that $(C(\mathcal{B}_{j_{s'}}) - C(\mathcal{B}_{j_s}))/(3h_n) \in \mathbb{Z}^d$. This is impossible since $C(\mathcal{B}_{j_s}) \in (0, 3h_n)^d$ for all s. Therefore, the sets $\mathcal{J}_s$ must be mutually disjoint. On the other hand, for every center $C(\mathcal{B}_\ell)$, there exists a center $C(\mathcal{B}_{j_s})$ in $(0, 3h_n)^d$ such that $(C(\mathcal{B}_\ell) - C(\mathcal{B}_{j_s}))/(3h_n) \in \mathbb{Z}^d$. Hence, $\bigcup_s \mathcal{J}_s = \mathcal{J}$.

Fix a $j \in \mathcal{J}$. Since the kernel K in the kernel density estimator has bandwidth h_n and support contained in $[-1,1]$, the point estimator $\widehat{f}_{\mathrm{KDE}}(\mathbf{x})$ only depends on the data points from the kernel data set $(\mathbf{X}'_\ell)_{\ell=1}^n$ that are in $NB(\mathcal{B}_j)$.

More generally, for two different indices $j, \widetilde{j} \in \mathcal{J}_s$, $j \neq \widetilde{j}$ and points $\mathbf{x}_1 \in \mathcal{B}_j$, $\mathbf{x}_2 \in \mathcal{B}_{\widetilde{j}}$, the point estimators $\widehat{f}_{\mathrm{KDE}}(\mathbf{x}_1)$ and $\widehat{f}_{\mathrm{KDE}}(\mathbf{x}_2)$ depend on $\{\mathbf{X}'_\ell : \mathbf{X}'_\ell \in NB(\mathcal{B}_j), \ell = 1, \ldots, n\}$ and $\{\mathbf{X}'_\ell : \mathbf{X}'_\ell \in NB(\mathcal{B}_{\widetilde{j}}), \ell = 1, \ldots, n\}$, respectively. The latter two sets are dependent if n is fixed (knowing that a data point is in one of the bins means that there can be at most $n-1$ in any of the other

bins). If we instead assume that the sample size of the data set $(\mathbf{X}'_\ell)_{\ell=1}^n$ is not n but $\mathcal{M}$ with $\mathcal{M} \sim \text{Poisson}(n)$, then $\{\mathbf{X}'_\ell : \mathbf{X}'_\ell \in A, \ell = 1, \dots, \mathcal{M}\}$ and $\{\mathbf{X}'_\ell : \mathbf{X}'_\ell \in B, \ell = 1, \dots, \mathcal{M}\}$ are independent, whenever A and B are disjoint sets. This will formally be shown in the proof of Lemma 7.2. Using Poisson point process theory, we also show in the proof of Lemma 7.2 that $\widehat{f}_{\text{KDE}}(\mathbf{x}_1)$ and $\widehat{f}_{\text{KDE}}(\mathbf{x}_2)$ are independent.

Proving oracle inequalities for the risk $R(\widetilde{f}, f_0) := \mathbb{E}_{f_0,\mathbf{X}}[(\widetilde{f}(\mathbf{X}) - f_0(\mathbf{X}))^2]$ in the standard i.i.d. setting typically first derives an oracle inequality for the empirical risk $\widehat{R}_n(\widehat{f}, f_0)$ as

$$\widehat{R}_n(\widehat{f}, f_0) := \mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n \big(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)^2\right].$$

Here empirical refers to the fact that the estimator $\widehat{f}$ is evaluated at the data points $\mathbf{X}_1, \dots, \mathbf{X}_n$. The derivation of an oracle inequality for the empirical risk can be further subdivided into several steps. The bound below refers to the step where our setting and the i.i.d. case differ the most. The proof (presented in Section 10) relies heavily on the construction of the bins above combined with Poissonization. Recall that $\varepsilon_i = Y_i - f(\mathbf{X}_i)$.

Lemma 7.2. *Consider the framework of Theorem 3.1. For any fixed $f \in \mathcal{F}$ and any $n \in \mathbb{N}, \delta > 0$ satisfying $\log^2(n)\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta)) \le n$,*

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n \varepsilon_i(\widehat{f}(\mathbf{X}_i) - f(\mathbf{X}_i))\right]\right| \\
&\le 2^{d+6}14e^2\|K\|_\infty^{2d}F^3 3^{\frac{7d}{2}} \\
&\cdot\left(\sqrt{\widehat{R}_n(\widehat{f}, f_0)}\log(n)\sqrt{\frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n}} + \log(n)\frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} + \delta\right) \\
&\quad + \frac{46F^2 2^d\|K\|_\infty^d}{n} + 8h_n^{2\beta}F^2 d^{2\beta}\|K\|_1^{2d} + \frac{\mathbb{E}_{\mathbf{X}}[(f_0(\mathbf{X}) - f(\mathbf{X}))^2]}{4} + \frac{\widehat{R}_n(\widehat{f}, f_0)}{4}.
\end{aligned}$$

With this lemma in place, we can prove in Section 10 the following bound on the empirical risk. This is similar to step (III) in the oracle inequality of Lemma 4 in [61].

Proposition 7.3. *Consider the framework of Theorem 3.1. For any fixed $f \in \mathcal{F}$ and any $n \in \mathbb{N}, \delta > 0$ satisfying $\log^2(n)\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta)) \le n$,*

$$\begin{aligned}
\widehat{R}_n(\widehat{f}, f_0) \le{}& \delta 2^{d+6}38e^2\|K\|_\infty^{2d}F^3 3^{\frac{9d}{2}} \\
&+ \frac{10}{3}\mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right] + \frac{8}{3}\Delta_n(\widehat{f}, f_0) \\
&+ 2^{d+6}38e^2\|K\|_\infty^{2d}F^3 3^{\frac{7d}{2}}\log(n)\frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} \\
&+ \frac{124F^2 2^d\|K\|_\infty^d}{n} + 22h_n^{2\beta}F^2 d^{2\beta}\|K\|_1^{2d} \\
&+ 4^{d+7}19^2e^4\|K\|_\infty^{4d}F^6 3^{7d}\log^2(n)\frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n}.
\end{aligned}$$

We now have all ingredients to finish the proof of Theorem 3.1.

Proof of Theorem 3.1. If $\log^2(n)\log(n\vee\mathcal{N}_{\mathcal{F}}(\delta))\geq n$, the statement follows with $C_1=4F^2$ by observing that $R(\widehat{f},f_0)\leq 4F^2$.

It remains to consider the case $\log^2(n)\log(n\vee\mathcal{N}_{\mathcal{F}}(\delta))\leq n$. The proof of Lemma 4, Part (I) in [61] states that for any $\epsilon\in(0,1]$,

$$\begin{aligned}&(1-\epsilon)\widehat{R}_n(\widehat{f},f_0)-\frac{F^2}{n\epsilon}\Big(15\log\big(\mathcal{N}_{\mathcal{F}}(\delta)\big)+75\Big)-26\delta F\\&\leq R(\widehat{f},f_0)\leq(1+\epsilon)\bigg(\widehat{R}_n(\widehat{f},f_0)+(1+\epsilon)\frac{F^2}{n\epsilon}\Big(12\log(\mathcal{N}_{\mathcal{F}}(\delta))+70\Big)+26\delta F\bigg).\end{aligned}\tag{7.2}$$

This lemma, derived for the standard nonparametric regression problem, relates the risk to its empirical counterpart. The inequality and its proof only depend on the $\mathbf{X}_i$ and on the function class $\mathcal{F}$, not on the noise or the response variables Y_i. Since in our regression model (2.3) the variables $\mathbf{X}_i$ are i.i.d. (the dependence is induced by the response variables Y_i and ε_i), this inequality is still valid.

Substituting the bound on $\widehat{R}_n(\widehat{f},f_0)$ from Proposition 7.3 in (7.2), choosing $\epsilon=1$ and f as a minimizer over $\mathcal{F}$ of $\mathbb{E}_{\mathbf{X}}\big[(f(\mathbf{X})-f_0(\mathbf{X}))^2\big]$, using the fact that $h_n\leq 2(\log(n)/n)^{1/d}$, and replacing the explicit constants by C_1,C_2,C_3 yields the result. □

7.2. *Proof of Theorem 3.4*

The following lemma provides a bound on the covering entropy.

Lemma 7.4 (Lemma 5 combined with Remark 1 of [61])**.** *For any $\delta>0$*

$$\log\big(\mathcal{N}_{\mathcal{F}(L,\mathbf{p},s,\infty)}(\delta)\big)\leq(s+1)\log\big(2^{2L+5}\delta^{-1}(L+1)p_0^2p_{L+1}^2s^{2L}\big).$$

The proof of Theorem 1 in [61] (see [62] for the precise statement) derives the following bound for the approximation error for function approximation in the function class $\mathcal{G}(q,\mathbf{d},\mathbf{t},\boldsymbol{\alpha},Q')$ by sparsely connected deep ReLU networks.

Theorem 7.5. *For every function $g\in\mathcal{G}(q,\mathbf{d},\mathbf{t},\boldsymbol{\alpha},Q')$ and whenever*

(i) $\sum_{i=1}^q\frac{\alpha_i+t_i}{2\alpha_i^*+t_i}\log_2(4t_i\vee 4\alpha_i)\log_2(n)\leq L\lesssim n\phi_n$,
(ii) $n\phi_n\lesssim\min_{i=1,\dots,L}p_i$,
(iii) $s\asymp n\phi_n\log(n)$,
(iv) $F\geq\max\{Q',1\}$,

then there exists a neural network $H\in\mathcal{F}(L,\mathbf{p},s,F)$ and a constant C_8 only depending on $q,\mathbf{d},\mathbf{t},\boldsymbol{\alpha},F$ and the implicit constants in (i), (ii) and (iii), such that

$$\|g-H\|_\infty^2\leq C_8\phi_n.$$

We now have all the necessary ingredients to prove Theorem 3.4

Proof of Theorem 3.4. We apply the general oracle inequality in Theorem 3.1 with the choice $\delta = n^{-1}$ to the neural network class $\mathcal{F}(L, \mathbf{p}, s, F)$ with parameter constraints as in the statement of the theorem. For the approximation error in the oracle inequality, we use Theorem 7.5. For the covering entropy, the bound from Lemma 7.4 gives $\log\left(n \vee \mathcal{N}_{\mathcal{F}(L,\mathbf{p},s,\infty)}(\frac{1}{n})\right) \lesssim (s+1)L\log(n) \asymp nL\phi_n \log^2(n)$. Since $L \gtrsim \log(n)$, we have $(\log(n)/n)^{2\beta/d} \lesssim L(n^{-2\beta/d} \vee n^{-1})$. As $\phi_n \gg n^{-1}$, $L\phi_n \log^4(n) + (\log(n)/n)^{2\beta/d} \lesssim L\max(\phi_n \log^4(n), n^{-2\beta/d})$. Thus, Theorem 3.1 yields

$$\begin{aligned} R(\widehat{f}_n, f_0) \leq\, & C_1 \frac{\log^2(n)\log(n \vee \mathcal{N}_{\mathcal{F}(L,\mathbf{p},s,\infty)}(\frac{1}{n}))}{n} + C_2\delta + C_3\left(\frac{\log(n)}{n}\right)^{\frac{2\beta}{d}} \\ & + 6\Delta_n(\widehat{f}_n, f_0) + 7 \inf_{f \in \mathcal{F}(L,\mathbf{p},s,F)} \mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right] \\ \leq\, & C_4 L \max\left(\phi_n \log^4(n), n^{-\frac{2\beta}{d}}\right) + 6\Delta_n(\widehat{f}_n, f_0), \end{aligned}$$

for a sufficiently large constant C_4, only depending on $q, \mathbf{d}, \boldsymbol{\alpha}, \mathbf{t}, F, \beta, K$ and the implicit constants in (ii), (iii), and (iv). This completes the proof. □

8. Proofs for Section 4

Lemma 8.1. *Let $m \leq m'$ be positive integers and $Q > 0$. Then $f : [-Q, Q]^{m'} \to \mathbb{R}$ with $f(\mathbf{x}) := \prod_{i=1}^m x_i$ is in $\mathcal{H}^\gamma_{m'}([-Q, Q]^{m'}, (Q+1)^m)$, for all $\gamma \geq m+1$.*

Proof. To compute the Hölder norm, it is sufficient to consider the function $g : [-Q, Q]^m \to \mathbb{R}$ with $g(x_1, \ldots, x_m) := \prod_{i=1}^m x_i$. Observe that $|\partial^0 g(\mathbf{x})| = |g(\mathbf{x})| \leq Q^m$, $\partial_{x_j} g(\mathbf{x}) = \prod_{i=1, i\neq j}^m x_i$ and $\partial_{x_j}\partial_{x_j} g = 0$, for $i = 1, \ldots, m$. This means that for all $\boldsymbol{\alpha} \in \mathbb{Z}^m_{\geq 0}$ it holds that $\partial^{\boldsymbol{\alpha}} g = 0$ if $\alpha_j \geq 2$ for some $j \in \{1, \ldots, m\}$. Rephrased, $\partial^{\boldsymbol{\alpha}} g \neq 0$ if and only if $\boldsymbol{\alpha} \in \{0, 1\}^m$. Furthermore for $\boldsymbol{\alpha} \in \{0, 1\}^m$, $|\partial^{\boldsymbol{\alpha}} g(\mathbf{x})| = |\prod_{i:\alpha_i = 0} x_i| \leq Q^{m - |\boldsymbol{\alpha}|_0}$, where $|\cdot|_0$ denotes the counting norm. There are $\binom{m}{m-|\boldsymbol{\alpha}|_0}$ ways to distribute $m - |\boldsymbol{\alpha}|_0$ zeros over a vector of length m. Hence for $\gamma \geq m+1$, we get by the binomial theorem

$$\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1 < \gamma} \|\partial^{\boldsymbol{\alpha}} g\|_\infty \leq \sum_{k=0}^m \binom{m}{k} Q^k = (Q+1)^m.$$

If $|\boldsymbol{\alpha}|_1 > m$, then there exists at least one j such that $\alpha_j \geq 2$ implying that $\partial^{\boldsymbol{\alpha}} g = 0$ in this case. In the case that $|\boldsymbol{\alpha}|_1 = m$, then either there exists a j such that $\alpha_j \geq 2$, so $\partial^{\boldsymbol{\alpha}} g = 0$, or $\boldsymbol{\alpha}$ is the vector with only ones, in which case $\partial^{\boldsymbol{\alpha}} g = 1$. Hence, $\gamma \geq m+1$ yields

$$\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1 = \lfloor\gamma\rfloor} \sup_{\mathbf{x},\mathbf{y} \in [-Q,Q]^m,\, \mathbf{x}\neq\mathbf{y}} \frac{|\partial^{\boldsymbol{\alpha}} g(\mathbf{x}) - \partial^{\boldsymbol{\alpha}} g(\mathbf{y})|}{|\mathbf{x} - \mathbf{y}|_\infty^{\gamma - \lfloor\gamma\rfloor}} = 0.$$

Together with the definition of the Hölder ball in (3.1), the statement follows. □

Proof of Lemma 4.1. The function $g_0 = (g_{0,1}, \ldots, g_{0,|\mathcal{R}|})$ is given by $g_{0,I} = \psi_I$ for all $I \in \mathcal{R}$. From $\psi_I \in \mathcal{H}^\gamma_{r_I}([0,1]^{r_I}, Q)$ with $r_I \leq r$ and $|I| \leq r$, it follows that $t_0 = r$ and $\alpha_0 = \gamma$. The function $g_1(u_1, \ldots, u_{|\mathcal{R}|}) = \prod_{i=1}^{|\mathcal{R}|} u_i$ is the product of $|\mathcal{R}|$ different factors in $[-Q, Q]$. Applying Lemma 8.1 yields $g_1 \in \mathcal{H}^\zeta_{|\mathcal{R}|}([-Q,Q]^{|\mathcal{R}|}, (Q+1)^{|\mathcal{R}|})$ for all $\zeta \geq |\mathcal{R}| + 1$. So $t_1 = |\mathcal{R}|$ and α_1 is arbitrarily large. □

Proof of Lemma 4.2. Since f is α-Hölder smooth, there exists a constant Q such that $f \in \mathcal{H}^\alpha_d([0,1]^d, Q)$. Thus for any $k = 0, 1, \ldots, \lfloor \alpha \rfloor$,

$$\frac{\partial^k}{\partial x_j^k} f(\mathbf{x}) = \Big(\prod_{i \neq j} f_i(x_i) \Big) f_j^{(k)}(x_j). \tag{8.1}$$

Since $\prod_{i \neq j} f_i$ is a density on $[0,1]^{d-1}$, it is nonnegative and $C := \prod_{i \neq j} \|f_i\|_\infty > 0$, with $\|\cdot\|_\infty$ the supremum norm on $[0,1]^d$. Since f_j only depends on x_j and f is α-Hölder smooth, for any $k = 0, 1, \ldots, \lfloor \alpha \rfloor$,

$$\frac{Q}{C} \geq \frac{1}{C} \sup_{(x_1,\ldots,x_d) \in [0,1]^d} \left| \Big(\prod_{i \neq j} f_i(x_i) \Big) f_j^{(k)}(x_j) \right| \geq \frac{\prod_{i \neq j} \|f_i\|_\infty}{C} \left\| f_j^{(k)} \right\|_\infty = \left\| f_j^{(k)} \right\|_\infty. \tag{8.2}$$

Similarly, by the α-Hölder smoothness of f and (8.1),

$$\begin{aligned} \frac{Q}{C} &\geq \frac{1}{C} \sup_{\mathbf{x},\mathbf{y} \in [0,1]^d, \mathbf{x} \neq \mathbf{y}} \frac{\left| \frac{\partial^{\lfloor \alpha \rfloor}}{\partial x_j^{\lfloor \alpha \rfloor}} f(\mathbf{x}) - \frac{\partial^{\lfloor \alpha \rfloor}}{\partial x_j^{\lfloor \alpha \rfloor}} f(\mathbf{y}) \right|}{|\mathbf{x} - \mathbf{y}|_\infty^{\alpha - \lfloor \alpha \rfloor}} \\ &\geq \frac{\prod_{i \neq j} \|f_i\|_\infty}{C} \sup_{x,y \in [0,1], x \neq y} \frac{\left| f_j^{(\lfloor \alpha \rfloor)}(x) - f_j^{(\lfloor \alpha \rfloor)}(y) \right|}{|x - y|^{\alpha - \lfloor \alpha \rfloor}}. \end{aligned} \tag{8.3}$$

From (8.2) and (8.3), it follows that $f_j \in \mathcal{H}^\alpha_1([0,1], (\lfloor \alpha \rfloor + 1)Q/C)$. □

Proof of Lemma 4.3. For the compositional sparse classes defined in (3.3), we also interpret g_{ij} as a function $[a_i, b_i]^{t_i} \to [a_{i+1}, b_{i+1}]$ if g_{ij} depends on t_i variables.

The function $g_0 = (g_{0,1}, \ldots, g_{0,2d})$ is given by $g_{0,i}(x_i) = f_i(x_i)/\|f_i\|_\infty$ for $i = 1, \ldots, d$ and $g_{0,i}(x_{i-d}) = F_{i-d}(x_{i-d})$ for $i = d+1, \ldots, 2d$. Each of these functions is univariate, so $t_0 = 1$. Since F_{i-d} is the c.d.f. of f_{i-d}, it holds that $F_{i-d} \in \mathcal{H}_1^{\gamma_0+1}([0,1], Q+1)$. Thus, $\alpha_0 = \gamma_0$. The function $g_1 = (g_{1,1}, \ldots, g_{1,d+1})$ is the identity function $g_{1,i}(y_i) = y_i$ for $i = 1, \ldots, d$ and $g_{1,d+1}(\mathbf{v}) = c(v_{d+1}, \ldots, v_{2d})$, so $t_1 = d$. For $i = 1, \ldots, d$, the domain of $g_{1,i}$ is $[0,1]$, so $g_{1,i} \in \mathcal{H}_1^\gamma([0,1], 2)$, for all $\gamma \geq 2$. Moreover, by assumption, $g_{1,d+1} \in \mathcal{H}_d^{\gamma_c}([0,1]^d, Q_c)$. This means that the Hölder smoothness of $g_{1,i}$ can be chosen to be arbitrarily large and consequently $g_{1,d+1}$ has the smallest Hölder smoothness among the component functions of g_1. Thus, $\alpha_1 = \gamma_c$. Set $Q' := Q_c \vee 1$, then $g_2(u, y_1, \ldots, y_d) = (\prod_{i=1}^d \|f_i\|_\infty) u \prod_{i=1}^d y_i$ is the product of $d+1$ different factors in $[-Q', Q']^{d+1}$. Applying Lemma 8.1 yields $g_2 \in \mathcal{H}^\gamma_{d+1}([-Q', Q']^{d+1}, Q^d(Q'+1)^{d+1})$ for all $\gamma \geq d+2$. So, $t_2 = d+1$ and the smoothness index α_2 can be taken to be arbitrarily large. □

Proof of Lemma 4.4. The function $g_0 = (g_{0,1}, \dots, g_{0,2d})$ is given by $g_{0,i}(x_i) = f_i(x_i)/\|f_i\|_\infty$ for $i = 1, \dots, d$ and $g_{0,i}(x_{i-d}) = F_{i-d}(x_{i-d})$ for $i = d+1, \dots, 2d$. Recall that $f_i \in \mathcal{H}_1^\gamma([0,1], Q)$, for all $i = 1, \dots, d$. Since F_{i-d} is the c.d.f. of f_{i-d}, it holds that $F_{i-d} \in \mathcal{H}_1^{\gamma+1}([0,1], Q+1)$. So $t_0 = 1$ and $\alpha_0 = \gamma$.

The function $g_1 = (g_{1,1}, \dots, g_{1,d+(d-1)})$ satisfies $g_{1,i}(u_i) = u_i$ (the identity function) for $i = 1, \dots, d$. For f of the form (4.8) it holds that $g_{1,i}(u_{d+1}, u_{i+1}) = c_{1,i+1-d}(u_{d+1}, u_{i+1})$ for $i = d+1, \dots, d+(d-1)$ and for f of the form (4.9) we have that $g_{1,i}(u_i, u_{i+1}) = c_{i-d,i+1-d}(u_i, u_{i+1})$ for $i = d+1, \dots, d+(d-1)$. For $i = 1, \dots, d$, we can define $g_{1,i}$ on $[0,1]$. Since $g_{1,i}$ is in this case the identity, treating the cases $0 < \beta \le 1$ and $\beta > 1$ separately, we find $g_{1,i} \in \mathcal{H}_1^\beta([0,1], 3)$, for all $\beta > 0$. By definition all bivariate copula densities are in $\mathcal{H}_2^{\gamma_c}([0,1]^2, Q)$. This means that $t_1 = 2$ and $\alpha_1 = \gamma_c$.

To realize the function

$$g_2(u_1, \dots, u_d, y_1, \dots, y_{d-1}) = \Big(\prod_{i=1}^d \|f_i\|_\infty \Big) \prod_{k=1}^d u_k \prod_{j=1}^{d-1} y_j,$$

we need to multiply $2d-1$ inputs. Now, g_2 can be defined on $[0, Q \vee 1]^{2d-1}$. Invoking Lemma 8.1 and $\prod_{i=1}^d \|f_i\|_\infty \le Q^d$, it holds that $g_2 \in \mathcal{H}_{2d-1}^\zeta([0, Q \vee 1]^{2d-1}, (Q+1)^{3d-1})$ for all $\zeta \ge 2d$, so $t_2 = 2d-1$ and α_2 is arbitrarily large. □

8.1. *Proof of Theorem 4.5*

We work in the density estimation model as defined in Section 2 with mixture density $f_0 = \sum_{j=1}^r a_j f_j$, where $a_1, \dots a_r$ are non-negative mixture weights summing up to one, and densities $f_j \in \mathcal{C}_d^{\beta_j}([0,1]^d, Q) \cap \mathcal{G}(q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j, Q)$, for all $j = 1, \dots, r,$. Recall that $\alpha_{i,j}^* := \alpha_{i,j} \prod_{\ell=i+1}^{q_j} (\alpha_{\ell,j} \wedge 1)$, and $\phi_n^\star := \max_{j=1,\dots,r} \phi_{n,j}$, where

$$\phi_{n,j} := \max_{i=0,\dots,q_j} n^{-\frac{2\alpha_{i,j}^*}{2\alpha_{i,j}^* + t_{i,j}}}.$$

Lemma 8.2 (Approximation of mixtures)**.** *Whenever*

(i) $\max_{j=1,\dots,r} \sum_{i=1}^{q_j} \frac{\alpha_{i,j}+t_{i,j}}{2\alpha_{i,j}^*+t_{i,j}} \log_2(4t_{i,j} \vee 4\alpha_{i,j}) \log_2(n) \le L \lesssim n\phi_n^\star$,

(ii) $n\phi_n^\star \lesssim \min_{i=1,\dots,L} p_i$,

(iii) $s \asymp n\phi_n^\star \log(n)$,

(iv) $F \ge \max\{Q, 1\}$,

then, for n large enough, there exists a network $H \in \mathcal{F}(L, \mathbf{p}, s, F)$ *and a constant* C *only depending on* $(q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j)_{j=1}^r$, r, F *and the implicit constants in (i), (ii) and (iii) such that*

$$\Big\| \sum_{j=1}^r a_j f_j - H \Big\|_\infty^2 \le C\phi_n^\star.$$

Proof. For positive constants $c_L, c_p, c_{s\ell}, c_{su}$, let $L^\star$, $\mathbf{p}^\star = (p_0^\star, \dots, p_{L^\star+1}^\star)$, and $s^\star$ be such that

(i') $\max_{j=1,\dots,r} \sum_{i=1}^{q_j} \frac{\alpha_{i,j}+t_{i,j}}{2\alpha^*_{i,j}+t_{i,j}} \log_2(4t_{i,j} \vee 4\alpha_{i,j}) \log_2(n) \leq L^\star \leq c_L n\phi_n^\star$

(ii') $n\phi_n^\star \leq c_p \min_{i=1,\dots,L^\star} p_i^\star$

(iii') $c_{s\ell} n\phi_n^\star \log(n) \leq s^\star \leq c_{su} n\phi_n^\star \log(n)$.

For n large enough (depending on $c_L, c_p, c_{s\ell}$, $(q_j, \mathbf{t}_j, \boldsymbol{\alpha}_j)_{j=1}^r$, r), we have

(I) $c_L n\phi_n^\star \leq (c_{s\ell}/(2r)) n\phi_n^\star \log(n)$,

(II) $n\phi_n^\star > rc_p$,

(III) $\lfloor c_L n\phi_{n,j} \rfloor \geq \sum_{i=1}^{q_j} \frac{\alpha_{i,j}+t_{i,j}}{2\alpha^*_{i,j}+t_{i,j}} \log_2(4t_{i,j} \vee 4\alpha_{i,j}) \log_2(n)$, for all $j = 1, \dots, r$,

(IV) $(c_{s\ell}/(4r)) n\phi_{n,j} \log(n) \geq 1$, for all $j = 1, \dots, r$.

For $j = 1, \dots, r$ define $L_j := \min\{L^\star, \lfloor c_L n\phi_{n,j} \rfloor\}$, $p_{i,j} := \lfloor p_i^\star / r \rfloor$ (for $1 \leq i \leq L_j$), and $s_j := \lfloor s^\star \phi_{n,j}/(2r\phi_n^\star) \rfloor$. Recall that $\phi_n^\star = \max_{j=1,\dots,r} \phi_{n,j}$. Using the definition of L_j and (III) yields

$$\sum_{i=1}^{q_j} \frac{\alpha_{i,j}+t_{i,j}}{2\alpha^*_{i,j}+t_{i,j}} \log_2(4t_{i,j} \vee 4\alpha_{i,j}) \log_2(n) \leq L_j \leq c_L n\phi_{n,j}.$$

Using (ii'), (II), and the definitions of ϕ_n^* and $\mathbf{p}_j$, we get that

$$n\phi_{n,j} \leq n\phi_n^* \leq 2c_p r \min_{i=1,\dots,L_j} \lfloor p_i^*/r \rfloor = 2c_p r \min_{i=1,\dots,L_j} p_{i,j}.$$

From (IV), $v \leq 2v - 1$ for all $v \geq 1$, the definition $s_j = \lfloor s^\star \phi_{n,j}/(2r\phi_n^\star) \rfloor$, (iii'), and $\lfloor u \rfloor \geq u - 1$ for all $u \in \mathbb{R}$, it follows that

$$\frac{c_{s\ell}}{4r} n\phi_{n,j} \log(n) \leq \frac{c_{s\ell}}{2r} n\phi_{n,j} \log(n) - 1 \leq s_j \leq \frac{c_{su}}{2r} n\phi_{n,j} \log(n).$$

This means that for $j = 1, \dots, r$, the class $\mathcal{F}(L_j, \mathbf{p}_j, s_j, F)$ and the function $f_j \in \mathcal{C}_d^\beta([0,1]^d, Q) \cap \mathcal{G}(q_j, \mathbf{d}_j, \mathbf{t}_j, \boldsymbol{\alpha}_j, Q)$ satisfy the conditions of Theorem 7.5. Applying Theorem 7.5 gives us that for each $j = 1 \dots, r$ there exist a network $H_j \in \mathcal{F}(L_j, \mathbf{p}_j, s_j, F)$ such that $\|f_j - H_j\|_\infty^2 \leq C_{8,j}\phi_{n,j}$. Since a_j is in $[0,1]$, multiplying the last weight matrix of H_j with a_j yields a network $a_j H_j$ in the same network class as H_j such that $\|a_j f_j - a_j H_j\|_\infty^2 \leq C_{8,j}\phi_{n,j}$.

Whenever $L_j < L^\star$, we can synchronize the depth by adding additional layers with 1×1 weight matrices and weight parameters $= 1$, such that

$$a_j \sigma(H_j) = \sigma(a_j H_j) \in \mathcal{F}\big(L^*, (\mathbf{p}_j, \underbrace{1, \dots, 1}_{(L^\star - L_j) \text{ times}}), s_j + (L^\star - L_j), F\big).$$

Since f_j is a density, $f_j \geq 0$ and $\|a_j\sigma(H_j) - f_j\|_\infty \leq \|a_j H_j - f_j\|_\infty$. We write $\widetilde{\mathbf{p}}_j := (\mathbf{p}_j, 1, \dots, 1)$. Placing all these networks in parallel yields a network

$$H \in \mathcal{F}\Big(L^\star, \sum_{j=1}^r \widetilde{\mathbf{p}}_j, \sum_{j=1}^r \big(s_j + (L^\star - L_j)\big), F\Big),$$

such that

$$\left\|\sum_{j=1}^{r} a_j f_j - H\right\|_\infty^2 \le \left(\sum_{j=1}^{r} \|a_j f_j - a_j H_j\|_\infty\right)^2 \le \left(\sum_{j=1}^{r} \sqrt{C_{8,j}\phi_{n,j}}\right)^2 \le r^2 \max_{j=1,\dots,r} C_{8,j}\phi_{n,j}.$$

A network with width $\mathbf{p}$ and sparsity s can always be embedded in a larger network of the same depth with width $\widetilde{\mathbf{p}} \ge \mathbf{p}$ (inequalities between vectors should always be understood as componentwise inequalities) and network sparsity $\widetilde{s} \ge s$. Thus it remains to show that $\sum_{j=1}^r \widetilde{\mathbf{p}}_j \le \mathbf{p}^\star$ and $\sum_{j=1}^r \big(s_j + (L^\star - L_j)\big) \le s^\star$. First consider the width. Using the definitions of $p_{i,j}$ and $\widetilde{\mathbf{p}}_j$, we get for $i = 1, \dots, L^\star$ that $\sum_{j=1}^r \widetilde{p}_{i,j} \le r \max_{j=1,\dots,r} \widetilde{p}_{i,j} \le r \max\{p_i^\star/r, 1\}$. From (II) and (ii'), we get that $p_i^\star/r > 1$. Hence, $\sum_{j=1}^r \widetilde{\mathbf{p}}_j \le \mathbf{p}^\star$. Now consider the sparsity. By the definition of s_j it holds that $s_j \le s^\star/(2r)$. From (i') and (I), we get that $L^\star \le s^\star/(2r)$. Hence, $\sum_{j=1}^r \big(s_j + (L^\star - L_j)\big) \le \sum_{j=1}^r \big(s_j + L^\star\big) \le s^\star$. □

Proof of Theorem 4.5. The derivative of a sum is the sum of the derivatives. Furthermore $(a_1, \dots, a_r)$ are non-negative mixture weights that sum op to one. Since $f_j \in \mathcal{C}_d^\beta([0,1]^d, Q)$ for $j = 1, \dots, r$, this means that also $f_0 \in \mathcal{C}_d^\beta([0,1]^d, Q)$. The statement of the theorem now follows from taking $\delta = 1/n$ and the network class $\mathcal{F}(L, \mathbf{p}, s, F)$ as the function class in Theorem 3.1. For the approximation error in the oracle inequality, we use Lemma 8.2 and for the covering entropy the bound from Lemma 7.4. Arguing similarly as in the proof of Theorem 3.4, this yields the result. □

9. Proofs for Section 5

Proof of Lemma 5.1. To represent $f_j(x_j|x_i) = x_i h_j(x_j) + (1 - x_i)h_j(1 - x_j)$ as a composition $g_1 \circ g_0$, choose $g_0(x_i, x_j) = (x_i, h_j(x_j), h_j(1 - x_j))$. Clearly $t_0 = 1$. Since $[0,1] \ni x_i \mapsto x_i$ lies in $\mathcal{H}_1^\gamma([0,1], 2)$, for all $\gamma > 0$, we get that $\alpha_0 = \gamma_j$. The function g_1 is given by $g_1(x_i, y_1, y_2) = x_i y_1 + (1 - x_i)y_2$, so $t_1 = 3$. The partial derivatives are $\partial_{x_i} g_1 = y_1 - y_2$, $\partial_{y_1} g_1 = x_i$, $\partial_{y_2} g_1 = 1 - x_i$, $\partial_{x_i}\partial_{y_1} g_1 = 1$ and $\partial_{x_i}\partial_{y_2} g_1 = -1$. All other partial derivatives of g_1 vanish. Thus $g_1 \in \mathcal{H}_3^\gamma([0,1] \times [-Q, Q]^2, 4(Q+1))$, for all $\gamma > 3$, so α_1 is arbitrarily large. □

Proof of Lemma 5.2. To represent $f_j(x_j|x_i) = h_j\big(\max\{x_j - x_i/4, 0\}\big)$ as a composition $g_1 \circ g_0$, choose $g_0(x_j, x_i) = \max\{0, x_j - x_i/4\}$. The derivative of this function is discontinuous along the line $x_j - x_i/4 = 0$. Observe that $|\max(0, a) - \max(0, a + b)| \le |b|$, for all real numbers a, b. Hence

$$\frac{|g_0(x_j, x_i) - g_0(x_j + u, x_i + v)|}{\max(|u|, |v|)} \le \frac{|u - v/4|}{\max(|u|, |v|)} \le \frac{5}{4}.$$

Thus $g_0 \in \mathcal{H}_2^1([0,1]^2, 9/4)$, so $\alpha_0 = 1$. The function g_1 is given by $g_1(y) = h_j(y)$, thus $t_1 = 1$ and $\alpha_1 = \gamma_j$. □

10. Proofs for Section 7

Proof of Lemma 7.2. The random variable $\varepsilon_i = \widehat{f}_{\mathrm{KDE}}(\mathbf{X}_i) - f_0(\mathbf{X}_i)$ is not centered. The first step adds and subtracts $\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]$ to get the centered random variable $\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]$ instead. Together with the triangle inequality, this gives

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n \varepsilon_i(\widehat{f}(\mathbf{X}_i) - f(\mathbf{X}_i))\right]\right| \\
&\le \left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i))\right]\right| \\
&+ \left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(f_0(\mathbf{X}_i) - f(\mathbf{X}_i))\right]\right| \\
&+ \left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i](\widehat{f}(\mathbf{X}_i) - f(\mathbf{X}_i))\right]\right| \\
&=: (I) + (II) + (III).
\end{aligned} \tag{10.1}$$

By the tower rule, we can in (II) first condition the expectation on $\mathbf{X}_i$. Now $(II) = 0$ follows from

$$\begin{aligned}
&\mathbb{E}_{f_0}\Big[\left(\varepsilon_i - \mathbb{E}_{f_0}\left[\varepsilon_i|\mathbf{X}_i\right]\right)\left(f_0(\mathbf{X}_i) - f(\mathbf{X}_i)\right)\Big|\mathbf{X}_i\Big] \\
&= \left(\mathbb{E}_{f_0}\left[\varepsilon_i|\mathbf{X}_i\right] - \mathbb{E}_{f_0}\left[\varepsilon_i|\mathbf{X}_i\right]\right)\left(f_0(\mathbf{X}_i) - f(\mathbf{X}_i)\right) = 0.
\end{aligned}$$

For real numbers a_i, b_i, we have $(|a_i| - |b_i|/2)^2 \ge 0$ and therefore $|a_i b_i| \le a_i^2 + b_i^2/4$ as well as $\sum_i |a_i b_i| \le \sum_i a_i^2 + \frac{1}{4}\sum_i b_i^2$. Bringing first the absolute value inside the expectation and applying this inequality twice, once to the sequences $(2\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]/\sqrt{n})_i$ and $((\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i))/\sqrt{n})_i$ and once to the sequences $(2\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]/\sqrt{n})_i$ and $((f_0(\mathbf{X}_i) - f(\mathbf{X}_i))/\sqrt{n})_i$ yields

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i](\widehat{f}(\mathbf{X}_i) - f(\mathbf{X}_i))\right]\right| \\
&\overset{(i)}{=} \left|\mathbb{E}_{f_0}\left[\sum_{i=1}^n \frac{2\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]}{\sqrt{n}}\frac{(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i))}{\sqrt{n}}\right]\right. \\
&\left. + \mathbb{E}_{f_0}\left[\sum_{i=1}^n \frac{2\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]}{\sqrt{n}}\frac{(f_0(\mathbf{X}_i) - f(\mathbf{X}_i))}{\sqrt{n}}\right]\right| \\
&\le 8\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n (\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])^2\right] + \frac{1}{4}\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n (f_0(\mathbf{X}_i) - f(\mathbf{X}_i))^2\right] \\
&+ \frac{1}{4}\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n \left(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i)\right)^2\right] \\
&\overset{(ii)}{=} 8\mathbb{E}_{f_0}\left[(\mathbb{E}_{f_0}[\varepsilon_1|\mathbf{X}_1])^2\right] + \frac{\mathbb{E}_{\mathbf{X}}[(f_0(\mathbf{X}) - f(\mathbf{X}))^2]}{4} + \frac{\widehat{R}_n(\widehat{f}, f_0)}{4},
\end{aligned}$$

where for (i) we added and subtracted the same term and (ii) follows from the definition of $\widehat{R}_n(\widehat{f}, f_0)$ and the fact that the $\mathbf{X}_i$ are i.i.d., Proposition 10.1 gives $\mathbb{E}_{f_0}[(\mathbb{E}_{f_0}[\varepsilon_1|\mathbf{X}_1])^2] \leq h_n^{2\beta} F^2 d^{2\beta} \|K\|_1^{2d}$ and so

$$(III) \leq 8h_n^{2\beta} F^2 d^{2\beta} \|K\|_1^{2d} + \frac{\mathbb{E}_{\mathbf{X}}[(f_0(\mathbf{X}) - f(\mathbf{X}))^2]}{4} + \frac{\widehat{R}_n(\widehat{f}, f_0)}{4}. \tag{10.2}$$

It remains to bound

$$(I) = \left| \mathbb{E}_{f_0} \left[\frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i)) \right] \right|$$

in (10.1). Let us briefly outline the main ideas. A standard strategy to do this is to use that $\widehat{f} \in \mathcal{F}$ and bound

$$(I) \leq \mathbb{E}_{f_0} \left[\frac{2}{n} \sup_{f \in \mathcal{F}} \left| \sum_{i=1}^{n} (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(f(\mathbf{X}_i) - f_0(\mathbf{X}_i)) \right| \right].$$

The remaining step is then to get the supremum $\sup_{f \in \mathcal{F}}$ out of the expectation. This is the central problem in empirical process theory. Standard empirical process techniques consider a covering of $\mathcal{F}$. On each of the balls in the covering the expectation does not change much, such that one can replace the supremum by a maximum over the centers of the covering balls plus some remainder terms. To control the expectation of the maximum over the centers of the balls from the covering, one can now apply the union bound together with concentration bounds. While we will follow these steps, there are various technical challenges that occur because of the dependence in the data.

The covering number of $\mathcal{F}$ with supremum norm balls of radius $\delta > 0$ has been called $\mathcal{N}_{\mathcal{F}}(\delta)$. If $\mathcal{N}_{\mathcal{F}}(\delta) < n$, then one can add some balls with centers in $\mathcal{F}$ to the covering, to obtain a (not necessarily optimal) covering with

$$N = n \vee \mathcal{N}_{\mathcal{F}}(\delta)$$

balls. By assumption, the N centers $f_1, \ldots, f_N$ lie in $\mathcal{F}$. Choose $k^* \in \{1, \ldots, N\}$ such that

$$\|\widehat{f} - f_{k^*}\|_\infty = \min_{1 \leq \ell \leq N} \|\widehat{f} - f_\ell\|_\infty.$$

In particular, k^* is random. Define $(IV) := |\mathbb{E}_{f_0}[\frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(f_{k^*}(\mathbf{X}_i) - f_0(\mathbf{X}_i))]|$. This gives us that

$$\begin{aligned}
&\left| \mathbb{E}_{f_0} \left[\frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(\widehat{f}(\mathbf{X}_i) - f_0(\mathbf{X}_i)) \right] \right| \\
&\leq \left| \mathbb{E}_{f_0} \left[\frac{2}{n} \sum_{i=1}^{n} (\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(\widehat{f}(\mathbf{X}_i) - f_{k^*}(\mathbf{X}_i)) \right] \right| + (IV) \\
&\overset{(i)}{\leq} \mathbb{E}_{f_0} \left[\frac{2\delta}{n} \sum_{i=1}^{n} \left| \varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i] \right| \right] + (IV) \\
&\overset{(ii)}{\leq} 4\delta \|K\|_\infty^d 2^d F + (IV)
\end{aligned} \tag{10.3}$$

where for (i) we used the property of the δ cover and the triangle inequality, and for (ii) we used Proposition 10.2.

In the next step we split the term (IV) into two parts, to separate the case were the $\mathbf{X}_i$ used for the regression are distributed 'nicely' (the event A below) from the case, were we have an extreme concentration of data points $\mathbf{X}_i$ (the event A^c). The (bad) second case can be shown to have small probability. For the derivation, we use the bins $\mathcal{B}_j$ as defined in Section 7.

Define the set A_j as $A_j := \{\sum_{i=1}^n \mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} \leq 2^{d+3} F \log(n)\}$ and the set A as the intersection

$$A := \bigcap_{j \in \mathcal{J}} A_j. \tag{10.4}$$

The two-stage nonparametric density estimator chooses a bandwidth $h_n \leq 2(\log(n)/n)^{1/d}$. This is equivalent to $2^d \log(n) \geq n h_n^d$. Together with the union bound, it follows that

$$\begin{aligned}
\mathbb{P}_{f_0}(A^c) &\leq \sum_{j \in \mathcal{J}} \mathbb{P}_{f_0}(A_j^c) \\
&\leq \sum_{j \in \mathcal{J}} \mathbb{P}_{f_0}\left(\sum_{i=1}^n \mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} > (7F + F) n h_n^d \right) \\
&\overset{(iii)}{\leq} \sum_{j \in \mathcal{J}} \mathbb{P}_{f_0}\left(\sum_{i=1}^n \mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} > 7F n h_n^d + n p_j \right) \\
&= \sum_{j \in \mathcal{J}} \mathbb{P}_{f_0}\left(\sum_{i=1}^n \left(\mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} - p_j \right) > 7F n h_n^d \right) \\
&\leq \sum_{j \in \mathcal{J}} \mathbb{P}_{f_0}\left(\left| \sum_{i=1}^n \left(\mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} - p_j \right) \right| > 7F n h_n^d \right),
\end{aligned} \tag{10.5}$$

where for (iii) we used that $p_j = \int_{\mathcal{B}_j} f_0(\mathbf{x})\, d\mathbf{x} \leq F h_n^d$ is the probability that an observation falls into bin $\mathcal{B}_j$.

We now apply the moment version of Bernstein's inequality stated in Proposition 10.4 (i). For any $m = 1, \ldots$

$$\mathbb{E}_{f_0}\left[|\mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}}|^m \right] = \mathbb{E}_{f_0}\left[\mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} \right] = p_j.$$

Setting $U = 1$ and $v = nFh_n^d \geq np_j$, we get from Bernstein's inequality in Proposition 10.4 (i) that

$$\begin{aligned}
\mathbb{P}_{f_0}\left(\left| \sum_{i=1}^n \left(\mathbb{1}_{\{\mathbf{X}_i \in \mathcal{B}_j\}} - p_j \right) \right| > 7F n h_n^d \right) &\leq 2 \exp\left(- \frac{7^2 F^2 n^2 h_n^{2d}}{2n(F h_n^d + 7F h_n^d)} \right) \\
&= 2 \exp\left(- \frac{49}{16} F n h_n^d \right) \\
&\leq 2 \exp(-3 n h_n^d) \\
&\overset{(v)}{\leq} 2 n^{-3},
\end{aligned}$$

where for (v) we used that by construction of the two-stage nonparametric density estimator, $h_n^d \geq \log(n)/n$. Combined with (10.5), we find

$$\mathbb{P}_{f_0}(A^c) \leq 2 \sum_{j \in \mathcal{J}} n^{-3} \leq 2n^{-2},$$

where the last inequality holds because $n \geq 3 > e$ implies $|\mathcal{J}| = h_n^{-d} \leq n/\log n \leq n$.

With

$$\xi_k := \sum_{i=1}^{n} \big(\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]\big)\big(f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)\mathbb{1}_A$$

one can decompose (IV) as follows

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^{n}\big(\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]\big)\big(f_{k^*}(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)\right]\right| \\
&\leq \left|\mathbb{E}_{f_0}\left[\frac{2}{n}\xi_{k^*}\right]\right| + \left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^{n}\big(\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]\big)\big(f_{k^*}(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)\mathbb{1}_{A^c}\right]\right|.
\end{aligned} \tag{10.6}$$

Moving the absolute value inside, using that f_{k^*} and f_0 are bounded by F and applying the Cauchy-Schwarz inequality yields

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^{n}(\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i])(f_{k^*}(\mathbf{X}_i) - f_0(\mathbf{X}_i))\mathbb{1}_{A^c}\right]\right| \\
&\leq \frac{4F}{n}\sum_{i=1}^{n}\mathbb{E}_{f_0}\left[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|\,\mathbb{1}_{A^c}\right] \\
&\leq \frac{4F}{n}\sum_{i=1}^{n}\sqrt{\mathbb{E}_{f_0}\left[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|^2\right]}\sqrt{\mathbb{P}_{f_0}(A^c)} \\
&\overset{(*)}{\leq} \frac{4F\sqrt{2(65F^2 2^{2d}\|K\|_\infty^{2d})}}{n} \\
&\leq \frac{46F^2 2^d \|K\|_\infty^d}{n}.
\end{aligned} \tag{10.7}$$

where for $(*)$ we used Proposition 10.3 and that $\mathbb{P}_{f_0}(A^c) \leq 2n^{-2}$, and for the last inequality we used that $4\sqrt{130} \leq 46$.

It remains to bound the term $|\mathbb{E}_{f_0}[\frac{2}{n}\xi_{k^*}]|$. Let as before $N = n \vee \mathcal{N}_{\mathcal{F}}(\delta)$, set

$$z_k := \sqrt{\log(N)} \vee \sqrt{n}\|f_k - f_0\|_n, \tag{10.8}$$

and define z_{k^*} as z_k for $k = k^*$. The empirical norm of a function g is

$$\|g\|_n := \left(\frac{1}{n}\sum_{i=1}^{n}(g(\mathbf{X}_i))^2\right)^{\frac{1}{2}}.$$

Using that k^* is the index of the center of the ball of the δ-cover closest to $\widehat{f}$, it holds that

$$\frac{z_{k^*}}{\sqrt{n}} = \sqrt{\frac{\log(N)}{n}} \vee \|f_{k^*} - f_0\|_n \leq \|\widehat{f} - f_0\|_n + \delta + \sqrt{\frac{\log(N)}{n}}.$$

Together with the Cauchy-Schwarz inequality, we obtain

$$\begin{aligned}\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\xi_{k^*}\right]\right| &\leq \frac{2}{\sqrt{n}}\mathbb{E}_{f_0}\left[\left|\frac{\xi_{k^*}}{\sqrt{n}}\right|\right] \\ &\leq \frac{2}{\sqrt{n}}\mathbb{E}_{f_0}\left[\frac{\|\widehat{f} - f_0\|_n + \delta + \sqrt{\frac{\log(N)}{n}}}{\frac{z_{k^*}}{\sqrt{n}}}\left|\frac{\xi_{k^*}}{\sqrt{n}}\right|\right] \\ &\leq 2\frac{\sqrt{\widehat{R}_n(\widehat{f}, f_0)} + \delta + \sqrt{\frac{\log(N)}{n}}}{\sqrt{n}}\sqrt{\mathbb{E}_{f_0}\left[\frac{\xi_{k^*}^2}{z_{k^*}^2}\right]}.\end{aligned} \tag{10.9}$$

For notational ease, define

$$C_{i,k} := \frac{f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)}{nh_n^d z_k}\mathbb{1}_A. \tag{10.10}$$

Since probabilities are always upper bounded by one, we have for any $a > 0$ and any square integrable random variable T, $\mathbb{E}[T^2] = \int_0^\infty \mathbb{P}\left(T^2 \geq t\right)\, dt = \int_0^\infty \mathbb{P}\left(|T| \geq \sqrt{t}\right)\, dt \leq a + \int_a^\infty \mathbb{P}\left(|T| \geq \sqrt{t}\right)\, dt$. Therefore, for any $a > 0$,

$$\begin{aligned}\mathbb{E}_{f_0}[\xi_{k^*}^2/z_{k^*}^2 \,|\, \mathbf{X}_1, \ldots, \mathbf{X}_n] &\leq \mathbb{E}_{f_0}\big[\max_k \xi_k^2/z_k^2 \,|\, \mathbf{X}_1, \ldots, \mathbf{X}_n\big] \\ &\leq a + \int_a^\infty \mathbb{P}_{f_0}\big(\max_k |\xi_k/z_k| \geq \sqrt{t} \,\big|\, \mathbf{X}_1, \ldots, \mathbf{X}_n\big)\, dt.\end{aligned} \tag{10.11}$$

The ratio ξ_k/z_k can be rewritten as the sum $\sum_{\ell=1}^n h_k(\mathbf{X}'_\ell)$, where conditionally on $\mathbf{X}_1, \ldots, \mathbf{X}_n$,

$$u \mapsto h_k(u) = \sum_{i=1}^n \left(\prod_{r=1}^d K\left(\frac{u_r - X_{i,r}}{h_n}\right) - \int_{\mathbb{R}^d}\prod_{r=1}^d K\left(\frac{v_r - X_{i,r}}{h_n}\right) f_0(\mathbf{v})\, d\mathbf{v}\right) C_{i,k}$$

is a deterministic function. Now let $\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2, \ldots$ be i.i.d. random variables distributed as $\mathbf{X}$ and independent of the data. Let $\mathcal{M}$ be a Poisson(n) random variable independent of the data and of the $\widetilde{\mathbf{X}}_i$. By the union bound and Pois-

sonization (Lemma 3.2),

$$\begin{aligned}
&\mathbb{P}_{f_0}\big(\max_k |\xi_k/z_k| \geq \sqrt{t} \,\big|\, \mathbf{X}_1,\dots,\mathbf{X}_n\big)\\
&\leq N \max_k \mathbb{P}_{f_0}\big(|\xi_k/z_k| \geq \sqrt{t} \,\big|\, \mathbf{X}_1,\dots,\mathbf{X}_n\big)\\
&= N \max_k \mathbb{P}_{f_0}\bigg(\bigg|\sum_{\ell=1}^{n} h_k(\mathbf{X}'_\ell)\bigg| \geq \sqrt{t} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg) \qquad (10.12)\\
&\leq \sqrt{2e\pi n}\, N \max_k \mathbb{P}_{f_0}\bigg(\bigg|\sum_{\ell=1}^{\mathcal{M}} h_k(\widetilde{\mathbf{X}}_\ell)\bigg| \geq \sqrt{t} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg).
\end{aligned}$$

With $W(\mathbf{X}_i) := \sum_{\ell=1}^{\mathcal{M}} \prod_{r=1}^{d} K(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n})$, we can write

$$\sum_{\ell=1}^{\mathcal{M}} h_k(\widetilde{\mathbf{X}}_\ell) = \sum_{i=1}^{n} \big(W(\mathbf{X}_i) - \mathbb{E}_{f_0}\big[W(\mathbf{X}_i)|\mathbf{X}_i\big]\big)C_{i,k}. \qquad (10.13)$$

Next we rewrite the sum over i. For this we use the bins $\mathcal{B}_j$ and the index sets of bins $\mathcal{J}_s$ as defined in Section 7.1. Using that the bins are disjoint and that each bin is in exactly one of the 3^d index classes $\mathcal{J}_s$, we have $\sum_{i=1}^{n} = \sum_{s=1}^{3^d} \sum_{j\in\mathcal{J}_s} \sum_{\mathbf{X}_i\in\mathcal{B}_j}$. Here we use $\sum_{\mathbf{X}_i\in\mathcal{B}_j}$ as shorthand notation for $\sum_{1\leq i\leq n, \text{s.t.}\, \mathbf{X}_i\in\mathcal{B}_j}$. For non-negative random variables $U_1,\dots,U_m$, $\{U_1+\ldots+U_m \geq \sqrt{t}\} \subseteq \cup_{j=1}^{m}\{U_j \geq \sqrt{t}/m\}$ and by the union bound $\mathbb{P}(U_1+\ldots+U_m \geq \sqrt{t}) \leq m \cdot \max_{j=1,\dots,m} \mathbb{P}(U_j \geq \sqrt{t}/m)$. Combined with (10.13),

$$\begin{aligned}
&\mathbb{P}_{f_0}\bigg(\bigg|\sum_{\ell=1}^{\mathcal{M}} h_k(\widetilde{\mathbf{X}}_\ell)\bigg| \geq \sqrt{t} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg)\\
&\leq 3^d \max_{s=1,\dots,3^d}\\
&\mathbb{P}_{f_0}\bigg(3^d\bigg|\sum_{j\in\mathcal{J}_s}\sum_{\mathbf{X}_i\in\mathcal{B}_j} \big(W(\mathbf{X}_i) - \mathbb{E}_{f_0}\big[W(\mathbf{X}_i)|\mathbf{X}_i\big]\big)C_{i,k}\bigg| \geq \sqrt{t} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg).
\end{aligned}$$

Thus, (10.11), (10.12) and the previous display give for any $a > 0$,

$$\begin{aligned}
&\mathbb{E}_{f_0}\bigg[\frac{\xi_{k^*}^2}{z_{k^*}^2} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg]\\
&\leq a + \int_a^\infty N 3^d \sqrt{2e\pi n} \max_k \max_{s=1,\dots,3^d}\\
&\qquad \mathbb{P}_{f_0}\bigg(3^d\bigg|\sum_{j\in\mathcal{J}_s}\sum_{\mathbf{X}_i\in\mathcal{B}_j} \big(W(\mathbf{X}_i) - \mathbb{E}_{f_0}[W(\mathbf{X}_i)|\mathbf{X}_i]\big)C_{i,k}\bigg| \geq \sqrt{t} \,\bigg|\, \mathbf{X}_1,\dots,\mathbf{X}_n\bigg)\, dt.
\end{aligned} \qquad (10.14)$$

We will now apply Bernstein's inequality in the form of Proposition 10.4 (i) to the random variables $Z_j := Z_{j,k} := \sum_{\mathbf{X}_i\in\mathcal{B}_j} W(\mathbf{X}_i)C_{i,k}$. For that we show

first that, conditionally on $\mathbf{X}_1,\dots,\mathbf{X}_n$, the random variables $Z_j, j\in\mathcal{J}_s$ with fixed s are jointly independent. To see this, recall that $W(\mathbf{X}_i):=\sum_{\ell=1}^{\mathcal{M}}\prod_{r=1}^d K(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n})$. The kernel K has support in $[-1,1]$. By the definition of the neighborhood $NB(\mathcal{B}_j)$ in (7.1), Z_j only depends on the $\widetilde{\mathbf{X}}_1,\dots,\widetilde{\mathbf{X}}_n$ that fall into $NB(\mathcal{B}_j)$, that is, $Z_j=\sum_{\mathbf{X}_i\in\mathcal{B}_j}\sum_{\ell=1}^{\mathcal{M}}\prod_{r=1}^d K(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n})C_{i,k}\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}$. The variables $C_{i,k}$ defined in (10.10) depend on $\mathbf{X}_1,\dots,\mathbf{X}_n$ but not on $\widetilde{\mathbf{X}}_1,\widetilde{\mathbf{X}}_2,\dots$ Working conditionally on $\mathbf{X}_1,\dots,\mathbf{X}_n$ and interchanging the summations, we can write $Z_j=\sum_{\ell=1}^{\mathcal{M}} g_j(\widetilde{\mathbf{X}}_\ell)\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}$, for suitable real-valued functions $g_1,g_2,\dots$ Since the kernel K has support in $[-1,1]$, it follows from the definition of $\mathcal{J}_s$ that if two different indices j and $\widetilde{j}$ are both in $\mathcal{J}_s$, then $\{\mathbf{x}: g_j(\mathbf{x})\neq 0\}\cap\{\mathbf{x}: g_{\widetilde{j}}(\mathbf{x})\neq 0\}=\varnothing$.

Consider a random measure of the form $\mathrm{N}=\sum_{k\geq 1}\xi_k\delta_{\mathbf{V}_k}$, with $\mathbf{V}_k$ d-dimensional random vectors, ξ_k $\{0,1,\dots\}$-valued random variables, and δ_u the point measure at u. Such a random measure is called a point process on $\mathbb{R}^d$ if for every bounded subset $A\subseteq\mathbb{R}^d$, we have $\mathbb{P}(\mathrm{N}(A)<\infty)=1$. A Poisson point process N with intensity measure μ is a point process such that for any Borel set $A\subseteq\mathbb{R}^d$, $\mathrm{N}(A)$ follows a Poisson distribution with intensity parameter $\mu(A)$ (with the convention $\mathrm{N}(A)=0$ a.s. if $\mu(A)=\infty$) and for pairwise disjoint Borel sets $A_1,\dots,A_k\subseteq\mathbb{R}^d$, $\mathrm{N}(A_1),\dots,\mathrm{N}(A_k)$ are (jointly) independent.

For N a Poisson point process and bounded measurable functions $\rho_1,\dots,\rho_m:\mathbb{R}^d\to\mathbb{R}$ with pairwise disjoint and bounded support $\{\mathbf{x}\in\mathbb{R}^d:\rho_i(\mathbf{x})\neq 0\}$, $i=1,\dots,m$, the random variables $\int\rho_i\,d\mathrm{N}$, $i=1,\dots,m$ are jointly independent.

To see this, one can use that bounded measurable functions can be uniformly approximated by simple functions. Thus, for every i, there exists a sequence of simple functions $(\rho_i^{(T)})_{T\in\mathbb{N}}$ such that $\rho_i^{(T)}\to\rho_i$ uniformly as $T\to\infty$ and one can also choose the support of $\rho_i^{(T)}$ to be contained in the support of ρ_i. Write $\rho_i^{(T)}=\sum_{\ell=1}^{L_i^{(T)}} a_{i\ell}^{(T)}\mathbb{1}_{A_{i\ell}^{(T)}}$ for pairwise disjoint Borel sets $A_{i\ell}^{(T)}$ contained in the support of ρ_i. For any T, $(A_{i\ell}^{(T)})_{i,\ell}$ are pairwise disjoint sets, the random variables $(\mathrm{N}(A_{i\ell}^{(T)}))_{i,\ell}$ are thus independent, and so are the integrals $\int\rho_i^{(T)}\,d\mathrm{N}=\sum_{\ell=1}^{L_i^{(T)}} a_{i\ell}^{(T)}\mathrm{N}(A_{i\ell}^{(T)})$, $i=1,\dots,m$. Since the support of $\rho_i^{(T)}$ is contained in the support of ρ_i, $\rho_i^{(T)}\to\rho_i$ uniformly, and $\mathrm{N}(\{\mathbf{x}\in\mathbb{R}^d:\rho_i(\mathbf{x})\neq 0\})<\infty$ almost surely, we obtain $\int\rho_i^{(T)}\,d\mathrm{N}\to\int\rho_i\,d\mathrm{N}$ almost surely as $T\to\infty$. Thus $\int\rho_i\,d\mathrm{N}$, $i=1,\dots,m$ are jointly independent, as claimed.

It follows from Section 4.9 in [59], the fact that $\widetilde{\mathbf{X}}_\ell$ are i.i.d., and $\mathcal{M}\sim\mathrm{Poisson}(n)$, that $\widetilde{\mathrm{N}}:=\sum_{\ell=1}^{\mathcal{M}}\delta_{\widetilde{\mathbf{X}}_\ell}$ is a Poisson point process on $[0,1]^d$. The functions $g_j\mathbb{1}_{\{\cdot\in NB(\mathcal{B}_j)\}}$ are measurable and bounded with bounded disjoint supports and thus, the random variables

$$Z_j=\int g_j\mathbb{1}_{\{\cdot\in NB(\mathcal{B}_j)\}}\,d\widetilde{\mathrm{N}}=\sum_{\ell=1}^{\mathcal{M}} g_j(\widetilde{\mathbf{X}}_\ell)\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}},\quad j\in\mathcal{J}_s$$

are jointly independent.

To apply Bernstein's inequality, it remains to check that there exist U and v such that $\sum_{j\in\mathcal{J}_s}\mathbb{E}_{f_0}[|Z_j|^m|\mathbf{X}_1,\ldots,\mathbf{X}_n]\leq\frac{1}{2}m!U^{m-2}v$, for $m=2,3,\ldots$

We have conditionally on $\mathbf{X}_i$ that

$$
\begin{aligned}
&\mathbb{E}_{f_0}\Big[\Big|\sum_{\mathbf{X}_i\in\mathcal{B}_j}W(\mathbf{X}_i)C_{i,k}\Big|^m\,\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big] \qquad (10.15)\\
&=\mathbb{E}_{f_0}\Big[\Big|\sum_{\mathbf{X}_i\in\mathcal{B}_j}\Big(\sum_{\ell=1}^{\mathcal{M}}\prod_{r=1}^{d}K\Big(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n}\Big)\Big)C_{i,k}\Big|^m\,\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big]\\
&\overset{(i)}{\leq}\mathbb{E}_{f_0}\Big[\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}\Big(\sum_{\ell=1}^{\mathcal{M}}\Big|\prod_{r=1}^{d}K\Big(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n}\Big)\Big|\Big)|C_{i,k}|\Big)^m\,\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big]\\
&\overset{(ii)}{=}\mathbb{E}_{f_0}\Big[\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}\Big(\sum_{\ell=1}^{\mathcal{M}}\Big|\prod_{r=1}^{d}K\Big(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n}\Big)\Big|\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\Big)|C_{i,k}|\Big)^m\\
&\quad\times\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big]\\
&\overset{(iii)}{\leq}\mathbb{E}_{f_0}\Big[\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}\Big(\sum_{\ell=1}^{\mathcal{M}}\|K\|_\infty^d\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\Big)|C_{i,k}|\Big)^m\,\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big]\\
&\overset{(iv)}{=}\mathbb{E}_{f_0}\Big[\Big(\sum_{\ell=1}^{\mathcal{M}}\|K\|_\infty^d\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\Big)^m\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}|\Big)^m\,\Big|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\Big]\\
&\overset{(v)}{=}\|K\|_\infty^{dm}\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}|\Big)^m\mathbb{E}_{f_0}\Big[\Big(\sum_{\ell=1}^{\mathcal{M}}\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\Big)^m\Big].
\end{aligned}
$$

Where (i) follows from the triangle inequality. For (ii) we used that $\mathbf{X}_i\in\mathcal{B}_j$ and that K has support in $[-1,1]$, so if $\widetilde{\mathbf{X}}_\ell$ is outside $NB(\mathcal{B}_j)$ then $\prod_{r=1}^{d}K(\frac{\widetilde{X}_{\ell,r}-X_{i,r}}{h_n})=0$. For (iii) we use that $\|K\|_\infty<\infty$ and that all terms are non-negative. The equality (iv) follows from observing that $\sum_{\ell=1}^{\mathcal{M}}\|K\|_\infty^d\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}$ does not depend on i and can be taken out of the sum. Finally (v) follows by taking all the constants out of the expectation, recalling that $C_{i,k}$ is $\sigma(\mathbf{X}_1,\ldots,\mathbf{X}_n)$-measurable and noting that $\sum_{\ell=1}^{\mathcal{M}}\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}$ is independent of $\mathbf{X}_1,\ldots,\mathbf{X}_n$.

Since $\widetilde{\mathbf{X}}_\ell$ are i.i.d. and $\mathcal{M}\sim\text{Poisson}(n)$, we have $\sum_{\ell=1}^{\mathcal{M}}\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\sim\text{Poisson}(n\widetilde{p}_j)$, where $\widetilde{p}_j$ denotes the probability that $\mathbf{X}\in NB(\mathcal{B}_j)$. Expressing the moments of the Poisson distribution as Bell polynomials [3] gives

$$
\mathbb{E}_{f_0}\Big[\Big(\sum_{\ell=1}^{\mathcal{M}}\mathbb{1}_{\{\widetilde{\mathbf{X}}_\ell\in NB(\mathcal{B}_j)\}}\Big)^m\Big]=\sum_{t=0}^{m}(n\widetilde{p}_j)^t\left\{{m\atop t}\right\}\leq(n\widetilde{p}_j\vee1)^m\sum_{t=0}^{m}\left\{{m\atop t}\right\},
$$

where $\left\{{m\atop t}\right\}$ denote the Stirling numbers of the second kind. The m-th Bell number equals the sum $\sum_{t=0}^{m}\left\{{m\atop t}\right\}$. Applying now the bound on Bell numbers

derived in Theorem 2.1 of [9] gives

$$\sum_{t=0}^{m} \left\{ {m \atop t} \right\} \leq \left(\frac{m}{\log(m+1)} \right)^m .$$

Due to $m \geq 2$, $\log(m+1) \geq \log(3) > 1$ and the right hand side of the previous display can be upper bounded by m^m. Using Stirling's formula ([60]) again, we get that $\sqrt{2\pi m} m^m e^{-m} \leq m!$. Since $m \geq 2$, we have $\sqrt{2\pi m} \geq e$ and thus $m^m \leq m! e^{m-1}$. Hence

$$\mathbb{E}_{f_0}\left[\left(\sum_{\ell=1}^{\mathcal{M}} \mathbb{1}_{\{\tilde{\mathbf{X}}_\ell \in NB(\mathcal{B}_j)\}} \right)^m \right] \leq m! e^{m-1} (n\widetilde{p}_j \vee 1)^m \leq m! e^{m-1} (F 3^d n h_n^d)^m .$$

The last inequality follows from observing that $\widetilde{p}_j \leq F 3^d h_n^d$ (the upper bound on f_0 times the Lebesgue measure of $NB(\mathcal{B}_j)$) and that $3^d F n h_n^d \geq 3^d F \log(n) \geq 1$. Combined with (10.15), this leads to

$$\begin{aligned} &\mathbb{E}_{f_0}\left[\left| \sum_{\mathbf{X}_i \in \mathcal{B}_j} W(\mathbf{X}_i) C_{i,k} \right|^m \middle| \mathbf{X}_1, \ldots, \mathbf{X}_n \right] \\ &\leq m! e^{m-1} (F 3^d n h_n^d)^m \|K\|_\infty^{dm} \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} |C_{i,k}| \right)^m . \end{aligned}$$

The previous inequality suggests to take the parameters v and U in Bernstein's inequality as upper bounds of $\sum_{j \in \mathcal{J}_s} (e\|K\|_\infty^d)^2 (F 3^d n h_n^d)^2 (\sum_{\mathbf{X}_i \in \mathcal{B}_j} |C_{i,k}|)^2$ and $\max_j e\|K\|_\infty^d 3^d F n h_n^d \sum_{\mathbf{X}_i \in \mathcal{B}_j} |C_{i,k}|$, respectively. To find a convenient expression for v, observe that

$$\begin{aligned} &\sum_{j \in \mathcal{J}_s} (e\|K\|_\infty^d)^2 (F 3^d n h_n^d)^2 \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} |C_{i,k}| \right)^2 \\ &= \sum_{j \in \mathcal{J}_s} (e 3^d \|K\|_\infty^d F)^2 n^2 h_n^{2d} \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} \left| \frac{(f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i))}{n h_n^d z_k} \mathbb{1}_A \right| \right)^2 \\ &= \sum_{j \in \mathcal{J}_s} \frac{(e 3^d \|K\|_\infty^d F)^2}{z_k^2} \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} |f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)| \, \mathbb{1}_A \right)^2 . \end{aligned}$$

By the Cauchy-Schwarz inequality,

$$\begin{aligned} \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} |f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)| \mathbb{1}_A \right)^2 &\leq \left(\mathbb{1}_A \sum_{\mathbf{X}_i \in \mathcal{B}_j} 1^2 \right) \left(\sum_{\mathbf{X}_i \in \mathcal{B}_j} \big(f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)^2 \right) \\ &\leq 2^{d+3} F \log(n) \sum_{\mathbf{X}_i \in \mathcal{B}_j} \big(f_k(\mathbf{X}_i) - f_0(\mathbf{X}_i)\big)^2 , \end{aligned}$$

where for the last inequality we used that the definition of the event A in (10.4) implies $\sum_{\mathbf{X}_i \in \mathcal{B}_j} 1 \leq 2^{d+3} F \log(n)$. By (10.8), $z_k \geq \sqrt{n} \|f_k - f_0\|_n$. Moreover,

$\sum_{i=1}^{n} = \sum_{j\in\mathcal{J}_s}\sum_{\mathbf{X}_i\in\mathcal{B}_j}$ and thus

$$\begin{aligned}
&\sum_{j\in\mathcal{J}_s}(e\|K\|_\infty^d)^2(F3^d nh_n^d)^2\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}|\Big)^2\\
&\le \sum_{j\in\mathcal{J}_s}\frac{(e3^d\|K\|_\infty^d F)^2}{n\|f_k-f_0\|_n^2}2^{d+3}F\log(n)\Big(\sum_{\mathbf{X}_i\in\mathcal{B}_j}(f_k(\mathbf{X}_i)-f_0(\mathbf{X}_i))^2\Big)\\
&= 2^{d+3}\frac{(e3^d\|K\|_\infty^d)^2}{n\|f_k-f_0\|_n^2}F^3\log(n)n\|f_k-f_0\|_n^2\\
&= 2^{d+3}(e3^d\|K\|_\infty^d)^2F^3\log(n).
\end{aligned}$$

Hence we can take $v = 2^{d+3}(e3^d\|K\|_\infty^d)^2F^3\log(n)$ in Bernstein's inequality.

To obtain a convenient expression for the U in Bernstein's inequality, we now bound $\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}|$. Using that by (10.8), $z_k \ge \sqrt{\log(N)}$, that f_k and f_0 are bounded by F, and that on the event A, $\sum_{\mathbf{X}_i\in\mathcal{B}_j}1 \le 2^{d+3}F\log(n)$ gives

$$\begin{aligned}
\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}| = \sum_{\mathbf{X}_i\in\mathcal{B}_j}\frac{|f_k(\mathbf{X}_i)-f_0(\mathbf{X}_i)|}{nh_n^d z_k}\mathbb{1}_A &\le \frac{2F}{nh_n^d\sqrt{\log(N)}}\sum_{\mathbf{X}_i\in\mathcal{B}_j}\mathbb{1}_A\\
&\le \frac{2^{d+4}F^2\log(n)}{nh_n^d\sqrt{\log(N)}}.
\end{aligned}$$

Hence it holds that

$$e\|K\|_\infty^d 3^d F nh_n^d\sum_{\mathbf{X}_i\in\mathcal{B}_j}|C_{i,k}| \le \frac{2^{d+4}e\|K\|_\infty^d 3^d F^3\log(n)}{\sqrt{\log(N)}}.$$

The support of the kernel is contained in $[-1,1]$. This means that $1 \le 2\|K\|_\infty$ and consequently, $e3^d\|K\|_\infty^d \ge 2$. Thus, setting $U = v/\sqrt{\log(N)}$ with $v = 2^{d+3}(e3^d\|K\|_\infty^d)^2F^3\log(n)$, as above, we obtain

$$\sum_{j\in\mathcal{J}_s}\mathbb{E}_{f_0}\Big[\Big|\sum_{\mathbf{X}_i\in\mathcal{B}_j}W(\mathbf{X}_i)C_{i,k}\Big|^m\Big|\mathbf{X}_1,\dots,\mathbf{X}_n\Big] \le \frac{m!}{2}vU^{m-2},$$

for all $m = 2,3,\dots$. Consequently we can apply Bernstein's inequality with those choices for U and v, conditioned on $\mathbf{X}_1,\dots,\mathbf{X}_n$.

Applying Bernstein's inequality on the sum over the variables Z_j, with v and

the bound U as defined above, we get that

$$\begin{aligned}
&\mathbb{P}_{f_0}\left(3^d\left|\sum_{j\in\mathcal{J}_s}\sum_{\mathbf{X}_i\in\mathcal{B}_j}\big(W(\mathbf{X}_i)-\mathbb{E}_{f_0}[W(\mathbf{X}_i)|\mathbf{X}_i]\big)C_{i,k}\right|\geq\sqrt{t}\,\middle|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\right)\\
&=\mathbb{P}_{f_0}\left(\left|\sum_{j\in\mathcal{J}_s}\big(Z_j-\mathbb{E}_{f_0}[Z_j|\mathbf{X}_1,\ldots,\mathbf{X}_n]\big)\right|\geq 3^{-d}\sqrt{t}\,\middle|\,\mathbf{X}_1,\ldots,\mathbf{X}_n\right)\\
&\leq 2\exp\left(-\frac{t3^{-2d}}{2\left(v+U3^{-d}\sqrt{t}\right)}\right)\\
&=2\exp\left(-\frac{t3^{-2d}}{2v\left(1+3^{-d}\sqrt{t/\log(N)}\right)}\right).
\end{aligned}$$

If $t\geq 3^{2d}\log(N)$, the previous expression can be further bounded by

$$\leq 2\exp\left(-\frac{\sqrt{t\log(N)}3^{-d}}{4v}\right). \tag{10.16}$$

Observe that this gives us an upper bound that is the same for all collections of bins $\mathcal{J}_s$ and all cover centers k. Choosing $a=64v^2 3^{2d}\log(N)$ in (10.14) gives

$$\begin{aligned}
&\mathbb{E}_{f_0}\left[\frac{\xi_{k^*}^2}{z_{k^*}^2}\middle|\mathbf{X}_1,\ldots,\mathbf{X}_n\right]\\
&\overset{(i)}{\leq} 64v^2 3^{2d}\log(N)+2N3^d\sqrt{2e\pi n}\int_{64v^2 3^{2d}\log(N)}^{\infty}\exp\left(-\sqrt{t}\frac{\sqrt{\log(N)}3^{-d}}{4v}\right)dt\\
&\overset{(ii)}{=} 64v^2 3^{2d}\log(N)+4N3^d\sqrt{2e\pi n}(16v^2 3^{2d})\frac{(2\log(N)+1)}{\log(N)}\exp\left(-2\log(N)\right)\\
&\overset{(iii)}{\leq} 64v^2 3^{2d}\log(N)+1280v^2 3^{3d}\\
&\overset{(iv)}{\leq} (2^{d+5}10(e\|K\|_\infty^d)^2F^3\log(n))^2 3^{7d}\log(N),
\end{aligned}$$

where for (i) we used (10.16) combined with the observation that if $t\geq 64v^2 3^{2d}\log(N)$ then $t\geq 3^{2d}\log(N)$, since $v\geq 1$. For (ii) we used that $\int_{b^2}^{\infty}e^{-\sqrt{u}c}du=2\int_b^{\infty}se^{-sc}ds=2(bc+1)e^{-bc}/c^2$. For (iii) we used that $\log(N)\geq 1$ so $(2\log(N)+1)/\log(N)\leq 4$ and $N=n\vee\mathcal{N}_{\mathcal{F}}(\delta)\geq n$, $\sqrt{2e\pi}\leq 5$, $\log(N)\geq 1$. For (iv) we substituted $v=2^{d+3}(e3^d\|K\|_\infty^d)^2F^3\log(n)$ and used that $\sqrt{1344}=4\sqrt{84}$ and $\sqrt{84}\leq 10$.

Together with (10.9), this yields

$$\begin{aligned}
&\left|\mathbb{E}_{f_0}\left[\frac{2}{n}\xi_{k^*}\right]\right|\\
\leq&2\frac{\sqrt{\widehat{R}_n(\widehat{f},f_0)}+\delta+\sqrt{\frac{\log(N)}{n}}}{\sqrt{n}}\sqrt{\mathbb{E}_{f_0}\left[\frac{\xi_{k^*}^2}{z_{k^*}^2}\right]}
\end{aligned}$$

$$\leq 2\frac{\sqrt{\widehat{R}_n(\widehat{f},f_0)}+\delta+\sqrt{\frac{\log(N)}{n}}}{\sqrt{n}}\sqrt{\left(2^{d+5}10e^2\|K\|_\infty^{2d}F^3\log(n)\right)^2 3^{7d}\log(N)}$$
$$=\left(\sqrt{\widehat{R}_n(\widehat{f},f_0)}+\delta+\sqrt{\frac{\log(N)}{n}}\right)2^{d+6}10e^2\|K\|_\infty^{2d}F^3\log(n)\sqrt{\frac{3^{7d}\log(N)}{n}}.$$

Inserting this bound in (10.6) together with (10.7) gives a bound for (IV). Together with (10.3) and (10.2) and combining the terms with δ, using that by assumption $\log^2(n)\log(N)=\log^2(n)\log(n\vee\mathcal{N}_{\mathcal{F}}(\delta))\leq n$, finishes the proof. □

Proof of Proposition 7.3. Expanding the square yields

$$\begin{aligned}\left(\widehat{f}(\mathbf{X}_i)-f_0(\mathbf{X}_i)\right)^2&=\left(\widehat{f}(\mathbf{X}_i)-Y_i+Y_i-f_0(\mathbf{X}_i)\right)^2\\&=(\widehat{f}(\mathbf{X}_i)-Y_i)^2+2(\widehat{f}(\mathbf{X}_i)-Y_i)(Y_i-f_0(\mathbf{X}_i))+(Y_i-f_0(\mathbf{X}_i))^2.\end{aligned}$$

We use this identity to rewrite the definition $\widehat{R}_n(\widehat{f},f_0)=\mathbb{E}_{f_0}[\frac{1}{n}\sum_{i=1}^n(\widehat{f}(\mathbf{X}_i)-f_0(\mathbf{X}_i))^2]$. Applying moreover that for any fixed $f\in\mathcal{F}$, we have by definition of $\Delta_n(\widehat{f},f_0)$ that

$$\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n(Y_i-\widehat{f}(\mathbf{X}_i))^2\right]\leq\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n(Y_i-f(\mathbf{X}_i))^2\right]+\Delta_n(\widehat{f},f_0)$$

yields

$$\begin{aligned}&\widehat{R}_n(\widehat{f},f_0)\\&=\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n\left((\widehat{f}(\mathbf{X}_i)-Y_i)^2+2(\widehat{f}(\mathbf{X}_i)-Y_i)(Y_i-f_0(\mathbf{X}_i))+(Y_i-f_0(\mathbf{X}_i))^2\right)\right]\\&\leq\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n\left((f(\mathbf{X}_i)-Y_i)^2+2(\widehat{f}(\mathbf{X}_i)-Y_i)(Y_i-f_0(\mathbf{X}_i))+(Y_i-f_0(\mathbf{X}_i))^2\right)\right]\\&\quad+\Delta_n(\widehat{f},f_0)\\&=\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n\left((f(\mathbf{X}_i)-Y_i)^2+2(f(\mathbf{X}_i)-Y_i)(Y_i-f_0(\mathbf{X}_i))+(Y_i-f_0(\mathbf{X}_i))^2\right)\right]\\&\quad+\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n(Y_i-f_0(\mathbf{X}_i))(\widehat{f}(\mathbf{X}_i)-f(\mathbf{X}_i))\right]+\Delta_n(\widehat{f},f_0)\\&=\mathbb{E}_{f_0}\left[\frac{1}{n}\sum_{i=1}^n(f(\mathbf{X}_i)-f_0(\mathbf{X}_i))^2\right]+\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n(Y_i-f_0(\mathbf{X}_i))(\widehat{f}(\mathbf{X}_i)-f(\mathbf{X}_i))\right]\\&\quad+\Delta_n(\widehat{f},f_0)\\&=\mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X})-f_0(\mathbf{X}))^2\right]+\mathbb{E}_{f_0}\left[\frac{2}{n}\sum_{i=1}^n\varepsilon_i(\widehat{f}(\mathbf{X}_i)-f(\mathbf{X}_i))\right]+\Delta_n(\widehat{f},f_0),\end{aligned}$$

where for the last equality we used that the $\mathbf{X}_i$ are independent and have the same distribution as $\mathbf{X}$.

Combined with Lemma 7.2, this yields

$$\begin{aligned}
\widehat{R}_n(\widehat{f}, f_0) &\leq \mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right] \\
&+ \sqrt{\widehat{R}_n(\widehat{f}, f_0)} 2^{d+6} 14 e^2 \|K\|_\infty^{2d} F^3 \log(n) \sqrt{\frac{3^{7d} \log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n}} \\
&+ 2^{d+6} 14 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} \log(n) \frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} \\
&+ \delta 2^{d+6} 14 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} + \frac{46 F^2 2^d \|K\|_\infty^d}{n} \\
&+ 8 h_n^{2\beta} F^2 d^{2\beta} \|K\|_1^{2d} + \frac{\mathbb{E}_{\mathbf{X}}[(f_0(\mathbf{X}) - f(\mathbf{X}))^2]}{4} + \frac{\widehat{R}_n(\widehat{f}, f_0)}{4} + \Delta_n(\widehat{f}, f_0).
\end{aligned}$$

Rewriting this and upper bounding constants, yields

$$\begin{aligned}
\widehat{R}_n(\widehat{f}, f_0) \leq \frac{5}{3}\mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right] \\
+ \sqrt{\widehat{R}_n(\widehat{f}, f_0)} 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 \log(n) \sqrt{\frac{3^{7d} \log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n}} \\
+ 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} \log(n) \frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} \\
+ \delta 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} \\
+ \frac{62 F^2 2^d \|K\|_\infty^d}{n} + 11 h_n^{2\beta} F^2 d^{2\beta} \|K\|_1^{2d} + \frac{4}{3}\Delta_n(\widehat{f}, f_0).
\end{aligned}$$

For real numbers a, c, ρ, satisfying $|a| \leq 2\sqrt{a}c + \rho$, we have $|a| \leq 2\sqrt{a}c + \rho \leq \frac{1}{2}|a| + 2c^2 + \rho$ and thus $|a| \leq 2\rho + 4c^2$. Applying this inequality with $a = \widehat{R}_n(\widehat{f}, f_0)$,

$$c = 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 \log(n) \sqrt{\frac{3^{7d} \log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n}},$$

and

$$\begin{aligned}
\rho =& \delta 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} + \frac{62 F^2 2^d \|K\|_\infty^d}{n} \\
&+ 2^{d+6} 19 e^2 \|K\|_\infty^{2d} F^3 3^{\frac{7d}{2}} \log(n) \frac{\log(n \vee \mathcal{N}_{\mathcal{F}}(\delta))}{n} \\
&+ 11 h_n^{2\beta} F^2 d^{2\beta} \|K\|_1^{2d} + \frac{4}{3}\Delta_n(\widehat{f}, f_0) + \frac{5}{3}\mathbb{E}_{\mathbf{X}}\left[(f(\mathbf{X}) - f_0(\mathbf{X}))^2\right]
\end{aligned}$$

yields the result. □

Proposition 10.1. $|\mathbb{E}_{f_0}[\varepsilon_i | \mathbf{X}_i]| \leq h_n^\beta d^\beta \|K\|_1^d F$.

Proof. By the construction of the ε_i in (2.2) and (2.3), $\varepsilon_i = \widehat{f}_{\mathrm{KDE}}(\mathbf{X}_i) - f_0(\mathbf{X}_i)$. Using moreover the definition of the multivariate kernel density estimator in (2.1) and writing $|\mathbf{v}|^{\boldsymbol{\alpha}}$ for $|v_1|^{\alpha_1} \cdot \ldots \cdot |v_d|^{\alpha_d}$, we obtain

$$
\begin{aligned}
\left|\mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]\right| &= \left|\mathbb{E}_{f_0}\left[\frac{1}{nh_n^d}\sum_{\ell=1}^{n}\prod_{r=1}^{d}K\left(\frac{X'_{\ell,r}-X_{i,r}}{h_n}\right) - f_0(\mathbf{X}_i)\middle|\mathbf{X}_i\right]\right| \\
&\overset{(i)}{=} \left|\frac{1}{h_n^d}\int_{[0,1]^d} f_0(\mathbf{u})\prod_{r=1}^{d}K\left(\frac{u_r-X_{i,r}}{h_n}\right)d\mathbf{u} - f_0(\mathbf{X}_i)\right| \\
&\overset{(ii)}{=} \left|\int_{\mathbb{R}^d}\left(\prod_{r=1}^{d}K(v_r)\right) f_0(X_{i,1}+v_1h_n,\ldots,X_{i,d}+v_dh_n)\,d\mathbf{v} - f_0(\mathbf{X}_i)\right| \\
&\overset{(iii)}{=} \left|\int_{\mathbb{R}^d}\left(\prod_{r=1}^{d}K(v_r)\right)\Big(f_0(X_{i,1}+v_1h_n,\ldots,X_{i,d}+v_dh_n) - f_0(\mathbf{X}_i)\Big)\,d\mathbf{v}\right| \\
&\overset{(iv)}{=} \left|\int_{\mathbb{R}^d}\left(\prod_{r=1}^{d}K(v_r)\right)\right. \\
&\qquad \times\Bigg(\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1\le\lfloor\beta\rfloor-1,\boldsymbol{\alpha}\neq 0}\frac{(h_n\mathbf{v})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}(\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i) \\
&\qquad\quad \left.+\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor}\frac{(h_n\mathbf{v})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}(\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i+h_n\tau(\mathbf{v})\mathbf{v})\Bigg)\,d\mathbf{v}\right| \\
&\overset{(v)}{=} \left|\int_{\mathbb{R}^d}\left(\prod_{r=1}^{d}K(v_r)\right)\right. \\
&\qquad \left.\times\left(\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor}\frac{(h_n\mathbf{v})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}\big((\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i+h_n\tau(\mathbf{v})\mathbf{v}) - (\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i)\big)\right)d\mathbf{v}\right| \\
&\overset{(vi)}{\le} h_n^{\lfloor\beta\rfloor}\int_{\mathbb{R}^d}\left|\prod_{r=1}^{d}K(v_r)\right| \\
&\qquad \times\left(\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor}\frac{|\mathbf{v}|^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}\big|(\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i+h_n\tau(\mathbf{v})\mathbf{v}) - (\partial^{\boldsymbol{\alpha}}f_0)(\mathbf{X}_i)\big|\right)d\mathbf{v} \\
&\overset{(vii)}{\le} h_n^{\lfloor\beta\rfloor}\int_{[-1,1]^d}\left|\prod_{r=1}^{d}K(v_r)\right|\left(\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor}\frac{|\mathbf{v}|^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}\,|h_n\tau(\mathbf{v})\mathbf{v}|_\infty^{\beta-\lfloor\beta\rfloor}\,F\right)d\mathbf{v} \\
&\overset{(viii)}{\le} h_n^{\beta}F\int_{[-1,1]^d}\left|\prod_{r=1}^{d}K(v_r)\right|\sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor}\frac{1}{\boldsymbol{\alpha}!}\,d\mathbf{v} \\
&\overset{(ix)}{\le} h_n^{\beta}\|K\|_1^d d^{\beta}F.
\end{aligned}
$$

Here we used for (i) that the $\mathbf{X}'_\ell$ are i.i.d. and independent of $\mathbf{X}_i$. For (ii) we substituted the transformed variables $v_r = (u_r - X_{i,r})/h_n$ and used that f_0

vanishes outside $[0,1]^d$, since f_0 has support in $[0,1]^d$ and is continuous on $\mathbb{R}^d$. For (iii) we used that a kernel integrates to 1 and that $f_0(\mathbf{X}_i)$ is a constant with respect to the integration variables. Step (iv) applies $\lfloor\beta\rfloor$-order Taylor expansion, that is, for a suitable $\tau(\mathbf{v}) \in (0,1)$,

$$f_0(\mathbf{X}_i + h_n\mathbf{v}) = f_0(\mathbf{X}_i) + \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1 \le \lfloor\beta\rfloor - 1, \boldsymbol{\alpha}\neq 0} \frac{(h_n\mathbf{v})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}(\partial^{\boldsymbol{\alpha}} f_0)(\mathbf{X}_i) + \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1 = \lfloor\beta\rfloor} \frac{(h_n\mathbf{v})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}(\partial^{\boldsymbol{\alpha}} f_0)(\mathbf{X}_i + h_n\tau(\mathbf{v})\mathbf{v}),$$

see Theorem 2.2.5 in [35]. For (v) we used that K is a kernel of order $\lfloor\beta\rfloor$ and therefore $\int v^m K(v)\,dv = 0$ for all $m = 1,\dots,\lfloor\beta\rfloor$. For (vi) we used that $h_n^{\lfloor\beta\rfloor}$ appears in every term of the sum. Jensen's inequality and triangle inequality are moreover applied to move the absolute value inside the integral and the sum. For (vii) we used that f_0 is in the β-Hölder ball with radius F and that K has support contained in $[-1,1]$. For $(viii)$ we used that $|\tau(\mathbf{v})| \le 1$. To see (ix), observe that for the multinomial distribution with number of trials $\lfloor\beta\rfloor$ and d event probabilities $(1/d,\dots,1/d)$, we have

$$1 = \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor} \frac{\lfloor\beta\rfloor!}{\boldsymbol{\alpha}!}\Big(\frac{1}{d}\Big)^{\alpha_1}\cdot\ldots\cdot\Big(\frac{1}{d}\Big)^{\alpha_d} = \lfloor\beta\rfloor! d^{-\lfloor\beta\rfloor} \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor} \frac{1}{\boldsymbol{\alpha}!} \ge d^{-\beta} \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_1=\lfloor\beta\rfloor} \frac{1}{\boldsymbol{\alpha}!}.$$

□

Proposition 10.2. $\mathbb{E}_{f_0}[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|] \le F\|K\|_\infty^d 2^{d+1}$.

Proof. By definition, $\varepsilon_i = Y_i - f_0(\mathbf{X}_i)$. Together with conditioning on $\mathbf{X}_i$, triangle inequality and Jensen's inequality this yields

$$\begin{aligned}\mathbb{E}_{f_0}[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|] &= \mathbb{E}_{f_0}[|Y_i - \mathbb{E}_{f_0}[Y_i|\mathbf{X}_i]|] \\ &\le 2\mathbb{E}_{f_0}\left[\mathbb{E}_{f_0}[|Y_i||\mathbf{X}_i]\right] \\ &\le 2\sum_{\ell=1}^n \frac{1}{nh_n^d}\mathbb{E}_{f_0}\left[\mathbb{E}_{f_0}\left[\prod_{r=1}^d \left|K\left(\frac{X'_{\ell,r} - X_{i,r}}{h_n}\right)\right| \,\Big|\, \mathbf{X}_i\right]\right].\end{aligned} \tag{10.17}$$

Using that $\|f_0\|_\infty \le F$ and the kernel K is supported on $[-1,1]$, we get by substitution

$$\begin{aligned}\mathbb{E}_{f_0}\left[\prod_{r=1}^d \left|K\left(\frac{X'_{\ell,r} - X_{i,r}}{h_n}\right)\right| \,\Big|\, \mathbf{X}_i\right] &\le F\int_{\mathbb{R}^d}\prod_{r=1}^d \left|K\left(\frac{u_r - X_{i,r}}{h_n}\right)\right| d\mathbf{u} \\ &= Fh_n^d\int_{\mathbb{R}^d}\prod_{r=1}^d |K(v_r)|\,d\mathbf{v}\end{aligned}$$

$$\leq F\|K\|_\infty^d 2^d h_n^d.$$

□

Proposition 10.3. $\mathbb{E}_{f_0}\left[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|^2\right] \leq 65F^2 2^{2d}\|K\|_\infty^{2d}$.

Proof. By definition, $\varepsilon_i = Y_i - f_0(\mathbf{X}_i)$. For a non-negative random-variable T, it holds that $\mathbb{E}[T^2] = \int_0^\infty \mathbb{P}(T^2 \geq t)\,dt = \int_0^\infty \mathbb{P}(T \geq \sqrt{t})\,dt$. Therefore

$$\begin{aligned}\mathbb{E}_{f_0}\left[|\varepsilon_i - \mathbb{E}_{f_0}[\varepsilon_i|\mathbf{X}_i]|^2\right] &= \mathbb{E}_{f_0}\left[\left|Y_i - \mathbb{E}_{f_0}[Y_i|\mathbf{X}_i]\right|^2\right] \\ &= \mathbb{E}_{f_0}\left[\mathbb{E}_{f_0}\left[\left|Y_i - \mathbb{E}_{f_0}[Y_i|\mathbf{X}_i]\right|^2 \,\middle|\, \mathbf{X}_i\right]\right] \\ &= \mathbb{E}_{f_0}\left[\int_0^\infty \mathbb{P}_{f_0}\left(\left|Y_i - \mathbb{E}_{f_0}[Y_i|\mathbf{X}_i]\right| \geq \sqrt{t} \,\middle|\, \mathbf{X}_i\right) dt\right].\end{aligned}$$

The probability can also be written as

$$\begin{aligned}&\int_0^\infty \mathbb{P}_{f_0}\left(\left|Y_i - \mathbb{E}_{f_0}[Y_i|\mathbf{X}_i]\right| \geq \sqrt{t} \,\middle|\, \mathbf{X}_i\right) dt \\ &= \int_0^\infty \mathbb{P}_{f_0}\left(\left|\sum_{\ell=1}^n \left(\prod_{r=1}^d K\left(\frac{X'_{\ell,r} - X_{i,r}}{h_n}\right)\right.\right.\right. \\ &\qquad \left.\left.\left. - \int_{[0,1]^d} f_0(\mathbf{u}) \prod_{r=1}^d K\left(\frac{u_r - X_{i,r}}{h_n}\right) d\mathbf{u}\right)\right| \geq nh_n^d\sqrt{t} \,\middle|\, \mathbf{X}_i\right) dt.\end{aligned}$$

This is a sum of i.i.d. random variables minus their expectation (conditionally on $\mathbf{X}_i$). Using that $\|f_0\|_\infty \leq F$ and the kernel K is supported on $[-1,1]$, we get using substitution

$$\begin{aligned}\mathbb{E}_{f_0}\left[\prod_{r=1}^d K^2\left(\frac{X'_{\ell,r} - X_{i,r}}{h_n}\right) \,\middle|\, \mathbf{X}_i\right] &\leq F\int_{\mathbb{R}^d} \prod_{r=1}^d K^2\left(\frac{u_r - X_{i,r}}{h_n}\right) d\mathbf{u} \\ &= Fh_n^d \int_{\mathbb{R}^d} \prod_{r=1}^d K^2(v_r)\,d\mathbf{v} \\ &\leq F\|K\|_\infty^{2d} 2^d h_n^d.\end{aligned}$$

Applying the bounded variable version of Bernstein's inequality in Proposition 10.4 (ii) with $v = nF\|K\|_\infty^{2d}2^d h_n^d$ and $b = 3\|K\|_\infty^d$ (that is, $b/3 = \|K\|_\infty^d$), we get that

$$\begin{aligned}&\int_0^\infty \mathbb{P}_{f_0}\left(\left|\sum_{\ell=1}^n \left(\prod_{r=1}^d K\left(\frac{X'_{\ell,r} - X_{i,r}}{h_n}\right)\right.\right.\right. \\ &\left.\left.\left. - \int_{[0,1]^d} f_0(\mathbf{u}) \prod_{r=1}^d K\left(\frac{u_r - X_{i,r}}{h_n}\right) d\mathbf{u}\right)\right| \geq nh_n^d\sqrt{t}\right) dt\end{aligned}$$

$$
\begin{aligned}
&\leq \int_0^\infty 1 \wedge 2\exp\left(-\frac{n^2 h_n^{2d} t}{2(n\|K\|_\infty^{2d} F 2^d h_n^d + \|K\|_\infty^d n h_n^d \sqrt{t})}\right) dt \\
&= \int_0^\infty 1 \wedge 2\exp\left(-\frac{n h_n^d t}{2(\|K\|_\infty^{2d} F 2^d + \|K\|_\infty^d \sqrt{t})}\right) dt \\
&\overset{(*)}{\leq} F^2 2^{2d} \|K\|_\infty^{2d} + 2\int_{F^2 2^{2d}\|K\|_\infty^{2d}}^\infty \exp\left(-\frac{n h_n^d \sqrt{t}}{4\|K\|_\infty^d}\right) dt \\
&\overset{(**)}{=} F^2 2^{2d} \|K\|_\infty^{2d} + \frac{64\|K\|_\infty^{2d}\left(\frac{F2^d\|K\|_\infty^d n h_n^d}{4\|K\|_\infty^d} + 1\right)\exp\left(-\frac{F2^d\|K\|_\infty^d n h_n^d}{4\|K\|_\infty^d}\right)}{n^2 h_n^{2d}} \\
&\overset{(***)}{\leq} F^2 2^{2d} \|K\|_\infty^{2d} + 64\|K\|_\infty^{2d}(F2^d + 1) \\
&\overset{(****)}{\leq} 65 F^2 2^{2d} \|K\|_\infty^{2d},
\end{aligned}
$$

where we used for $(*)$ that $2(\|K\|_\infty^{2d} F 2^d + \|K\|_\infty^d \sqrt{t}) \leq 4\|K\|_\infty^d \sqrt{t}$ whenever $t \geq F^2 2^{2d}\|K\|_\infty^{2d}$. For $(**)$ we used that $\int_{b^2}^\infty e^{-a\sqrt{u}}\, du = 2\int_b^\infty s e^{-sa} ds = 2(ba+1)e^{-ba}/a^2$, with $a = nh_n^d/(4\|K\|_\infty^d)$ and $b = F2^d\|K\|_\infty^d$. For $(***)$, we used that $nh_n^d \geq \log(n) \geq 1$ and that $0 < \exp(-x) \leq 1$ for $x \geq 0$. For $(****)$, we used that $F2^d + 1 \leq 2F2^d \leq F^2 2^{2d} \leq F^2 2^{2d}$. The result follows from observing that $\mathbb{E}[c] = c$, for any real number c. □

Proposition 10.4. *Given independent random variables $Z_1, \ldots, Z_n$.*

(i) (moment version) If for some constants U and v the moment bounds $\sum_{i=1}^n \mathbb{E}[|Z_i|^m] \leq \frac{1}{2} m! U^{m-2} v$ hold for all $m = 2, 3, \ldots$, then

$$\mathbb{P}\left(\left|\sum_{i=1}^n (Z_i - \mathbb{E}[Z_i])\right| > t\right) \leq 2e^{-\frac{t^2}{2v+2Ut}}.$$

(ii) (bounded version) If for some constants b and v, the bounds $|Z_i| \leq b$ and $\sum_{i=1}^n \mathbb{E}[|Z_i|^2] \leq v$ hold for all $i = 1, \ldots, n$, then,

$$\mathbb{P}\left(\left|\sum_{i=1}^n (Z_i - \mathbb{E}[Z_i])\right| > t\right) \leq 2e^{-\frac{t^2}{2v+2bt/3}}.$$

These formulations of Bernstein's inequality are based on Corollary 2.11 and Equation (2.10) in [13].

Acknowledgments

We are extremely grateful for the detailed comments that we received from the two referees. One referee suggested several improvements including a more streamlined Poissonization argument in the proof of Lemma 7.2. We want to thank Claire Donnat for pointing us to Lindsey's method and we are grateful to Kaizheng Wang for inspiring discussions.

Funding

The research has been supported by the NWO/STAR grant 613.009.034b and the NWO Vidi grant VI.Vidi.192.021.

References

[1] Aas, K., Czado, C., Frigessi, A. and Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance Math. Econom.* **44** 182-198. MR2517884

[2] Agarwal, R., Chen, Z. and Sarma, S. V. (2017). A novel nonparametric maximum likelihood estimator for probability density functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39** 1294 – 1308.

[3] Ahle, T. D. (2022). Sharp and simple bounds for the raw moments of the binomial and Poisson distributions. *Statist. Probab. Lett.* **182** Paper No. 109306, 5. https://doi.org/10.1016/j.spl.2021.109306 MR4347487

[4] Allman, E. S., Matias, C. and Rhodes, J. A. (2009). Identifiability of parameters in latent structure models with many observed variables. *Ann. Statist.* **37** 3099–3132. https://doi.org/10.1214/09-AOS689 MR2549554

[5] Barron, A. R. (1993). Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inform. Theory* **39** 930–945. MR1237720

[6] Barron, A. R. (1994). Approximation and estimation bounds for artificial neural networks. *Machine learning* **14** 115–133.

[7] Bauer, B. and Kohler, M. (2019). On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *Ann. Statist.* **47** 2261–2285. https://doi.org/10.1214/18-AOS1747 MR3953451

[8] Bedford, T. and Cooke, R. M. (2001). Probability density decomposition for conditionally dependent random variables modeled by vines. *Ann. Math. Artif. Intell.* **32** 245–268. MR1859866

[9] Berend, D. and Tassa, T. (2010). Improved bounds on Bell numbers and on moments of sums of random variables. *Probab. Math. Statist.* **30** 185–205. MR2792580

[10] Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *J. Roy. Statist. Soc. Ser. B* **36** 192–236. MR373208

[11] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer. MR2247587

[12] Bos, T. and Schmidt-Hieber, J. (2023). Simulation-code: A supervised deep learning method for nonparametric density estimation. https://github.com/Bostjm/Simulation-code.

[13] Boucheron, S., Lugosi, G. and Massart, P. (2013). *Concentration inequalities*. Oxford University Press, Oxford. https://doi.org/10.1093/acprof:oso/9780199535255.001.0001 MR3185193

[14] Brown, L. D., Carter, A. V., Low, M. G. and Zhang, C.-H. (2004). Equivalence theory for density estimation, Poisson processes and Gaussian

white noise with drift. *Ann. Statist.* **32** 2074–2097. https://doi.org/10.1214/009053604000000012 MR2102503

[15] Chae, M. (2022). Rates of convergence for nonparametric estimation of singular distributions using generative adversarial networks. *arXiv e-prints* arXiv:2202.02890. https://doi.org/10.48550/arXiv.2202.02890

[16] Chen, M., Liao, W., Zha, H. and Zhao, T. (2020). Distribution approximation and statistical estimation guarantees of generative adversarial networks. *arXiv e-prints* arXiv:2002.03938. https://doi.org/10.48550/arXiv.2002.03938

[17] Cheng, P. E. (1994). Nonparametric estimation of mean functionals with data missing at random. *Journal of the American Statistical Association* **89** 81–87. https://doi.org/10.1080/01621459.1994.10476448

[18] Cherubini, U., Luciano, E. and Vecchiato, W. (2004). *Copula methods in finance. Wiley Finance Series*. John Wiley & Sons. MR2250804

[19] Czado, C. (2019). *Analyzing dependent data with vine copulas. Lecture Notes in Statistics* **222**. Springer. MR3931334

[20] Czado, C. and Nagler, T. (2022). Vine copula based modeling. *Annu. Rev. Stat. Appl.* **9** 453–477. https://doi.org/10.1146/annurev-statistics-040220-101153 MR4394916

[21] Dinh, L., Sohl-Dickstein, J. and Bengio, S. (2017). Density estimation using Real NVP. In *International Conference on Learning Representations*.

[22] Drouet Mari, D. and Kotz, S. (2001). *Correlation and dependence*. Imperial College Press, London; distributed by World Scientific Publishing Co., Inc., River Edge, NJ. https://doi.org/10.1142/9781860949753 MR1835042

[23] Dudley, R. M. (1984). A course on empirical processes. In *Ecole d'été de Probabilités de Saint-Flour XII-1982* 1–142. Springer. MR0876079

[24] Durante, F. and Sempi, C. (2010). Copula theory: an introduction. In *Copula theory and its applications. Lect. Notes Stat. Proc.* **198** 3–31. Springer, Heidelberg. https://doi.org/10.1007/978-3-642-12465-5_1 MR3051261

[25] Efron, B. and Tibshirani, R. (1996). Using specially designed exponential families for density estimation. *Ann. Statist.* **24** 2431–2461. https://doi.org/10.1214/aos/1032181161 MR1425960

[26] Gänssler, P. (1983). *Empirical processes. Institute of Mathematical Statistics Lecture Notes—Monograph Series* **3**. Institute of Mathematical Statistics, Hayward, CA. MR744668

[27] Gao, Z. and Hastie, T. (2022). LinCDE: conditional density estimation via Lindsey's method. *J. Mach. Learn. Res.* **23** Paper No. [52], 55. MR4420777

[28] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* 249–256. JMLR Workshop and Conference Proceedings.

[29] Grathwohl, W., Chen, R. T. Q., Bettencourt, J. and Duvenaud, D. (2019). Scalable reversible generative models with free-form con-

tinuous dynamics. In *International Conference on Learning Representations*.

[30] Gutmann, M. U. and Hyvärinen, A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research* **13** 307–361. MR2913702

[31] Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The elements of statistical learning*, second ed. *Springer Series in Statistics*. Springer, New York. https://doi.org/10.1007/978-0-387-84858-7 MR2722294

[32] Heckerman, E. and Nathwani, N. (1992). Toward normative expert systems: Part II. Probability-based representations for efficient knowledge acquisition and inference. *Methods of Information in medicine* **31** 106–116.

[33] Huynh, H. T. and Nguyen, L. (2020). Nonparametric maximum likelihood estimation using neural networks. *Pattern Recognition Letters* **138** 580-586. https://doi.org/10.1016/j.patrec.2020.09.006

[34] Johnson, N. L. and Kotz, S. (1977). On some generalized Farlie-Gumbel-Morgenstern distributions-II Regression, correlation and further generalizations. *Communications in Statistics-Theory and Methods* **6** 485–496. MR0438589

[35] Kantorovitz, S. (2016). *Several real variables. Springer Undergraduate Mathematics Series*. Springer, [Cham]. https://doi.org/10.1007/978-3-319-27956-5 MR3467258

[36] Kasahara, H. and Shimotsu, K. (2014). Non-parametric identification and estimation of the number of components in multivariate mixtures. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **76** 97–111. https://doi.org/10.1111/rssb.12022 MR3153935

[37] Kennedy, E. H., Ma, Z., McHugh, M. D. and Small, D. S. (2017). Non-parametric methods for doubly robust estimation of continuous treatment effects. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **79** 1229–1245. https://doi.org/10.1111/rssb.12212 MR3689316

[38] Kingma, D. P. and Welling, M. (2019). An introduction to variational autoencoders. *Foundations and Trends in Machine Learning* **12** 307-392. https://doi.org/10.1561/2200000056

[39] Kohler, M. and Krzyżak, A. (2005). Adaptive regression estimation with multilayer feedforward neural networks. *J. Nonparametr. Stat.* **17** 891–913. MR2192165

[40] Kohler, M. and Langer, S. (2021). On the rate of convergence of fully connected deep neural network regression estimates. *Ann. Statist.* **49** 2231–2249. MR4319248

[41] Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques. Adaptive Computation and Machine Learning series*. MIT Press. MR2778120

[42] Korb, K. B. and Nicholson, A. E. (2010). *Bayesian Artificial Intelligence, Second Edition. Chapman & Hall/CRC Computer Science & Data Analysis*. CRC Press. MR3100449

[43] Künzel, S. R., Sekhon, J. S., Bickel, P. J. and Yu, B. (2019). Metalearners for estimating heterogeneous treatment effects using machine

learning. *Proceedings of the National Academy of Sciences* **116** 4156-4165. https://doi.org/10.1073/pnas.1804597116

[44] Lauritzen, S. L. (1996). *Graphical models. Oxford Statistical Science Series* **17**. The Clarendon Press, Oxford University Press, New York Oxford Science Publications. MR1419991

[45] Lindsey, J. K. (1974). Construction and comparison of statistical models. *J. Roy. Statist. Soc. Ser. B* **36** 418–425. MR365794

[46] Lindsey, J. K. (1974). Comparison of probability distributions. *J. Roy. Statist. Soc. Ser. B* **36** 38–47. MR362643

[47] Marzouk, Y., Ren, Z., Wang, S. and Zech, J. (2023). Distribution learning via neural differential equations: a nonparametric statistical perspective. *arXiv e-prints* arXiv:2309.01043. https://doi.org/10.48550/arXiv.2309.01043

[48] Mörters, P. and Peres, Y. (2010). *Brownian motion. Cambridge Series in Statistical and Probabilistic Mathematics* **30**. Cambridge University Press, Cambridge With an appendix by Oded Schramm and Wendelin Werner. https://doi.org/10.1017/CBO9780511750489 MR2604525

[49] Moschopoulos, P. and Staniswalis, J. G. (1994). Estimation given conditionals from an exponential family. *Amer. Statist.* **48** 271–275. https://doi.org/10.2307/2684831 MR1321892

[50] Murphy, K. P. (2012). *Machine Learning: a Probabilistic Perspective*. MIT press.

[51] Nagler, T. and Czado, C. (2016). Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *J. Multivariate Anal.* **151** 69–89. MR3545278

[52] Nelsen, R. B. (2007). *An introduction to copulas. Springer Series in Statistics*. Springer. MR2197664

[53] Nussbaum, M. (1996). Asymptotic equivalence of density estimation and Gaussian white noise. *Ann. Statist.* **24** 2399–2430. MR1425959

[54] Oko, K., Akiyama, S. and Suzuki, T. (2023). Diffusion models are minimax optimal distribution estimators. In *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato and J. Scarlett, eds.). *Proceedings of Machine Learning Research* **202** 26517–26582. PMLR.

[55] Onken, D., Wu Fung, S., Li, X. and Ruthotto, L. (2021). OT-Flow: Fast and accurate continuous normalizing flows via optimal transport. *Proceedings of the AAAI Conference on Artificial Intelligence* **35** 9223-9232.

[56] Pearl, J. (2009). *Causality*. Cambridge University Press. MR2548166

[57] Poggio, T., Mhaskar, H., Rosasco, L., Miranda, B. and Liao, Q. (2017). Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. *International Journal of Automation and Computing* **14** 503–519.

[58] Ray, K. and Schmidt-Hieber, J. (2018). The Le Cam distance between density estimation, Poisson processes and Gaussian white noise. *Math. Stat. Learn.* **1** 101–170. https://doi.org/10.4171/msl/1-2-1 MR4050245

[59] Resnick, S. (1992). *Adventures in stochastic processes*. Birkhäuser Boston,

Inc., Boston, MA. MR1181423
[60] Robbins, H. (1955). A remark on Stirling's formula. *Amer. Math. Monthly* **62** 26–29. MR0069328
[61] Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with ReLU activation function. *Ann. Statist.* **48** 1875–1897. MR4134774
[62] Schmidt-Hieber, J. and Vu, D. (2024). Correction to "Nonparametric regression using deep neural networks with ReLU activation function". *The Annals of Statistics* **52** 413 – 414. https://doi.org/10.1214/24-AOS2351 MR4718422
[63] Schmidt-Hieber, J. and Zamolodtchikov, P. (2022). Local convergence rates of the least squares estimator with applications to transfer learning. *arXiv e-prints* arXiv:2204.05003. MR4746591
[64] Schreuder, N., Brunel, V.-E. and Dalalyan, A. (2021). Statistical guarantees for generative models without domination. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory* (V. Feldman, K. Ligett and S. Sabato, eds.). *Proceedings of Machine Learning Research* **132** 1051–1071. PMLR. MR4227353
[65] Scott, D. W. (2015). *Multivariate density estimation*, second ed. *Wiley Series in Probability and Statistics*. John Wiley & Sons, Inc., Hoboken, NJ Theory, practice, and visualization. MR3329609
[66] Silverman, B. W. (1986). *Density estimation for statistics and data analysis. Monographs on Statistics and Applied Probability*. Chapman & Hall, London. https://doi.org/10.1007/978-1-4899-3324-9 MR848134
[67] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.). *Proceedings of Machine Learning Research* **37** 2256–2265. PMLR, Lille, France.
[68] Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S. and Poole, B. (2021). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.
[69] Stéphanovitch, A., Aamari, E. and Levrard, C. (2023). Wasserstein GANs are minimax optimal distribution estimators. *arXiv e-prints* arXiv:2311.18613. https://doi.org/10.48550/arXiv.2311.18613 MR4829484
[70] Stöber, J., Joe, H. and Czado, C. (2013). Simplified pair copula constructions—limitations and extensions. *J. Multivariate Anal.* **119** 101–118. MR3061418
[71] van de Geer, S. A. (2000). *Applications of empirical process theory. Cambridge Series in Statistical and Probabilistic Mathematics* **6**. Cambridge University Press, Cambridge. MR1739079
[72] van Der Vaart, A. W. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer New York. MR1385671
[73] Vandermeulen, R. A. and Ledent, A. (2021). Beyond smoothness:

Incorporating low-rank analysis into nonparametric density estimation. In *Advances in Neural Information Processing Systems* (M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang and J. W. Vaughan, eds.) **34** 12180–12193. Curran Associates, Inc.

[74] Wand, M. P. and Jones, M. C. (1994). *Kernel smoothing*. Chapman and Hall. MR1319818

[75] Wang, H. and Kim, J. K. (2023). Statistical inference using regularized M-estimation in the reproducing kernel Hilbert space for handling missing data. *Ann. Inst. Statist. Math.* **75** 911–929. https://doi.org/10.1007/s10463-023-00872-8 MR4655783

[76] Wang, K. (2023). Pseudo-labeling for kernel ridge regression under covariate shift. *arXiv e-prints* arXiv:2302.10160. https://doi.org/10.48550/arXiv.2302.10160