Learning with tree tensor networks: complexity estimates and model selection

BERTRAND MICHELI, ANTHONY NOUY

1 Centrale Nantes, Laboratoire de Mathématiques Jean Leray, CNRS UMR 6629, France. E-mail: "bertrand.michel@ec-nantes.fr; anthony.nouy@ec-nantes.fr

In this paper, we propose and analyze a model selection method for tree tensor networks in an empirical risk minimization framework and analyze its performance over a wide range of smoothness classes. Tree tensor networks, or tree-based tensor formats, are prominent model classes for the approximation of high-dimensional functions in numerical analysis and data science. They correspond to sum-product neural networks with a sparse connectivity associated with a dimension partition tree $T$, widths given by a tuple $r$ of tensor ranks, and multilinear activation functions (or units). The approximation power of these model classes has been proved to be optimal (or near to optimal) for classical smoothness classes. However, in an empirical risk minimization framework with a limited number of observations, the dimension tree $T$ and ranks $r$ should be selected carefully to balance estimation and approximation errors. In this paper, we propose a complexity-based model selection strategy à la Barron, Birgé, Massart. Given a family of model classes associated with different trees, ranks, tensor product feature spaces and sparsity patterns for sparse tensor networks, a model is selected by minimizing a penalized empirical risk, with a penalty depending on the complexity of the model class. After deriving bounds of the metric entropy of tree tensor networks with bounded parameters, we deduce a form of the penalty from bounds on suprema of empirical processes. This choice of penalty yields a risk bound for the predictor associated with the selected model. In a least-squares setting, after deriving fast rates of convergence of the risk, we show that the proposed strategy is (near to) minimax adaptive to a wide range of smoothness classes including Sobolev or Besov spaces (with isotropic, anisotropic or mixed dominating smoothness) and analytic functions. We discuss the role of sparsity of the tensor network for obtaining optimal performance in several regimes. In practice, the amplitude of the penalty is calibrated with a slope heuristics method. Numerical experiments in a least-squares regression setting illustrate the performance of the strategy for the approximation of multivariate functions and univariate functions identified with tensors by tensorization (quantization).

Keywords: tensor networks, statistical learning, metric entropy, model selection, minimax adaptive.

1. Introduction

Typical tasks in statistical learning include the estimation of a regression function or of posterior probabilities for classification (supervised learning), or the estimation of the probability distribution of a random variable from samples of the distribution (unsupervised learning). These approximation tasks can be formulated as a minimization problem
of a risk functional $\mathcal{R}(f)$ whose minimizer $f^*$ is the target (or oracle) function, and such that $\mathcal{R}(f) - \mathcal{R}(f^*)$ measures some discrepancy between the function $f$ and $f^*$. The risk is usually defined as

$$\mathcal{R}(f) = \mathbb{E}(\gamma(f, Z)),$$

with $Z = (X, Y)$ for supervised learning or $Z = X$ for unsupervised learning, and where $\gamma$ is a contrast function. For supervised learning, the contrast $\gamma$ is usually chosen as $\gamma(f, (x, y)) = \ell(y, f(x))$ where $\ell(y, f(x))$ measures some discrepancy between $y$ and the prediction $f(x)$ for a given realization $(x, y)$ of $(X, Y)$. In practice, given i.i.d. realizations $(Z_1, \ldots, Z_n)$ of $Z$, an approximation $\hat{f}_n^M$ is obtained by the minimization of an empirical risk

$$\hat{R}_n(f) = \frac{1}{n} \sum_{i=1}^{n} \gamma(f, Z_i)$$

over a subset of functions $M$, also called a model class or hypothesis set. Assuming that the risk admits a minimizer $f^M$ over $M$, the error $\mathcal{R}(\hat{f}_n^M) - \mathcal{R}(f^*)$ can be decomposed into two contributions: an approximation error $\mathcal{R}(f^M) - \mathcal{R}(f^*)$ which quantifies the best we can expect from the model class $M$, and an estimation error $\mathcal{R}(\hat{f}_n^M) - \mathcal{R}(f^M)$ which is due to the use of a limited number of observations. For a given model class, a first problem is to understand how these errors behave under some assumptions on the target function. When considering an increasing sequence of model classes, the approximation error decreases but the estimation error usually increases. Then strategies are required for the selection of a particular model class.

In many applications, the target function $f^*(x)$ is a function of many variables $x = (x_1, \ldots, x_d)$. For applications in image or signal classification, $x$ may be an image (with $d$ the number of pixels or patches) or a discrete time signal (with $d$ the number of time instants) and $f^*(x)$ provides a label to a particular input $x$. For applications in computational science, the target function may be the solution of a high-dimensional partial differential equation, a parameter-dependent equation or a stochastic equation. In all these applications, when $d$ is large and when the number of observations is limited, one has to rely on suitable model classes $M$ of moderate complexity that exploit specific structures of the target function $f^*$ and yield an approximation $\hat{f}_n^M$ with low approximation and estimation errors. Typical examples of model classes include additive functions $f_1(x_1) + \cdots + f_d(x_d)$, sums of multiplicative functions $\sum_{k=1}^{m} f_1^k(x_1) \cdots f_d^k(x_d)$, projection pursuit $f_1(w_1^T x) + \cdots + f_m(w_m^T x)$, or feed-forward neural networks $\sigma_L \circ f_L \circ \cdots \circ \sigma_1 \circ f_1(x)$ where the $f_k$ are affine maps and the $\sigma_k$ are given nonlinear functions.

In this paper, we consider the class of functions in tree-based tensor format, or tree tensor networks. These model classes are well-known approximation tools in numerical analysis and computational physics and have also been more recently considered in statistical learning. They are particular cases of feed-forward neural networks with an architecture given by a dimension partition tree and multilinear activation functions (see [31, 13]). For an overview of these tools, the reader is referred to the monograph [25] and
the surveys [36, 6, 30, 11, 12]. Some results on the approximation power of tree tensor networks can be found in [38, 23, 5] for multivariate functions, or in [29, 28, 1, 2, 3] for tensorized (or quantized) functions.

A tree-based tensor format is a set of functions

\[ M^T_r(V) = \{ f \in V : \text{rank}_\alpha(f) \leq r_\alpha, \alpha \in T \}, \]

where \( T \) is a dimension partition tree over \( \{1, \ldots, d\} \), \( r = (r_\alpha) \in \mathbb{N}^{\mid T\mid} \) is a tuple of integers and \( V = V_1 \otimes \ldots \otimes V_d \) is a finite dimensional tensor space of multivariate functions (e.g., polynomials, splines), that is a tensor product feature space. A function \( f \) in \( M^T_r(V) \) is such that for each \( \alpha \in T \), the \( \alpha \)-rank \( \text{rank}_\alpha(f) \) of \( f \) is bounded by \( r_\alpha \). That means that for each \( \alpha \in T \), \( f \) admits a representation

\[ f(x) = \sum_{k=1}^{r_\alpha} g^\alpha_k(x_\alpha) h^\alpha_k(x_{\alpha^c}) \]

for some functions \( g^\alpha_k \) and \( h^\alpha_k \) of complementary groups of variables. Such a representation can be written using tensor diagram notations as

\[ f(x) = \begin{array}{c}
g^\alpha \\
\downarrow \\
x_{\alpha}
\end{array} \begin{array}{c}
h^\alpha \\
\downarrow \\
x_{\alpha^c}
\end{array} \]

where \( g^\alpha \) and \( h^\alpha \) are order-two tensors with indices \((k, x_\alpha)\) and \((k, x_{\alpha^c})\) respectively, and the edge between the two tensors has to be interpreted as a contraction of the two connected tensors. A function \( f \) in \( M^T_r(V) \) admits a parametrization in terms of a collection of low-order tensors \( v = (v^\alpha)_{\alpha \in T} \) forming a tree tensor network. For instance, for the dimension tree of Figure 1a, the function \( f \) admits the representation of Figure 1b using tensor diagram notations. If the tensors \( v^\alpha \) are sparse, the tensor network \( v \) is called a sparse tensor network. By identifying the tensors \( v^\alpha \) with multilinear functions with values in \( \mathbb{R}^{r_\alpha} \), the function \( f \) also admits a representation as a composition of multilinear functions, that corresponds to a sum-product feed-forward neural network illustrated on Figure 2.

Model classes \( M^T_r(V) \) associated with different trees (or architecture of the tensor network) are known to capture very different structures of multivariate functions. The choice of a good tree is then crucial in many applications. This requires robust strategies that select not only the ranks for a given tree but the tree and the associated ranks.

The main contribution of the paper is a complexity-based strategy for the selection of a model class in an empirical risk minimization framework. Given a family \( (M_m)_{m \in M} \) of tensor networks (full or sparse) associated with different trees \( T_m \), ranks \( r_m \), feature tensor spaces \( V_m \) (and different sparsity patterns for sparse tensor networks), and given the corresponding predictors \( \hat{f}_m \) that minimize the empirical risk, we propose a strategy to select a particular model \( \hat{m} \) with a guaranteed performance. For that purpose, we make
use of the model selection approach of Barron, Birgé and Massart (see [34] for a general introduction to the topic) where \( \hat{m} \) is obtained by minimizing a penalized empirical risk

\[
\hat{R}_n(\hat{f}_m) + \text{pen}(m)
\]

with a penalty function \( \text{pen}(m) \) derived from complexity estimates of the model classes \( M_m \), of the form \( \text{pen}(m) \sim O(\sqrt{C_m/n}) \) (up to logarithmic terms) in a general setting, or of the form \( \text{pen}(m) \sim O(C_m/n) \) (again up to logarithmic terms) in a bounded least-squares setting where faster convergence rates can be obtained. Here, the complexity \( C_m \) is related to the number of parameters in the tensor network (total number of entries of the tensors \( v^\alpha \)), or the number of non-zero parameters in the tensor network when exploiting sparsity of the tensors \( v^\alpha \).

In a bounded least-squares setting (for regression or density estimation), using particular features based on tensorization of functions, we find that our strategy is minimax (or near to minimax) adaptive to a wide range of smoothness spaces including Sobolev or Besov spaces with isotropic, anisotropic or mixed dominating smoothness, and analytic function spaces.

In practice, the penalty is taken of the form \( \text{pen}(m) = \lambda \sqrt{C_m/n} \) (or \( \text{pen}(m) = \lambda C_m/n \) in a bounded least-squares setting), where \( \lambda \) is calibrated with the slope heuristics method proposed in [9]. The family of models can be generated by adaptive learning algorithms such as the ones proposed in [21, 20].
Learning with tree tensor networks

Note that our method is a $\ell_0$ type approach. Convex regularization methods would be an interesting alternative route to follow. A straightforward convexification of tensor formats consists in using the sum of nuclear norms of unfoldings (see e.g. [39] for Tucker format) but this is known to be far from optimal from a statistical point of view (see [37]). A convex regularization method based on the tensor nuclear norm has been proposed for the Tucker format, or shallow tensor network, which comes with theoretical guarantees (see [42]). However, there is no straightforward extension of this approach to general tree tensor networks.

The outline of the paper is as follows. In Section 2, we describe the model class of tree tensor networks (or tree-based tensor formats) [25, 18]. In Section 3, we provide estimates of the metric and bracketing entropies in $L^p$ spaces for tree tensor networks $M_m$ with bounded parameters. In Section 4, we derive bounds for the estimation error in a classical empirical risk minimization framework. These bounds are deduced from concentration inequalities for empirical processes. Then we present the complexity-based model selection approach and we derive risk bounds for particular choices of penalty in a general setting. We then introduce different collections of tensor networks (full or sparse) corresponding to different adaptive settings, where the feature space and the tree are considered either fixed or free, and we analyze the richness of these collections of models. Then in Section 5, we consider a bounded least-squares setting, for which we derive im-

Figure 2: A feed-forward sum-product neural network (a) corresponding to the format $M^T_r(V)$ with $N_v = 10$ features per variable $x_v$, the dimension tree $T$ of Figure 1a, and a tuple of ranks $r$ given in figure (b).

(a) Feed-forward neural network. (b) Ranks $r_\alpha$, $\alpha \in T$. 

Note that our method is a $\ell_0$ type approach. Convex regularization methods would be an interesting alternative route to follow. A straightforward convexification of tensor formats consists in using the sum of nuclear norms of unfoldings (see e.g. [39] for Tucker format) but this is known to be far from optimal from a statistical point of view (see [37]). A convex regularization method based on the tensor nuclear norm has been proposed for the Tucker format, or shallow tensor network, which comes with theoretical guarantees (see [42]). However, there is no straightforward extension of this approach to general tree tensor networks.

The outline of the paper is as follows. In Section 2, we describe the model class of tree tensor networks (or tree-based tensor formats) [25, 18]. In Section 3, we provide estimates of the metric and bracketing entropies in $L^p$ spaces for tree tensor networks $M_m$ with bounded parameters. In Section 4, we derive bounds for the estimation error in a classical empirical risk minimization framework. These bounds are deduced from concentration inequalities for empirical processes. Then we present the complexity-based model selection approach and we derive risk bounds for particular choices of penalty in a general setting. We then introduce different collections of tensor networks (full or sparse) corresponding to different adaptive settings, where the feature space and the tree are considered either fixed or free, and we analyze the richness of these collections of models. Then in Section 5, we consider a bounded least-squares setting, for which we derive im-
proved risk bounds with fast rates. That allows us in Section 6 to prove that our strategy is (near to) minimax adaptive to a large range of smoothness classes. In supplementary materials, we present the practical aspects of the model selection approach, which includes the slope heuristics method for penalty calibration and the exploration strategies for the generation of a sequence of model classes and associated predictors. Also, we present some numerical experiments that validate the proposed model selection strategy, and we give the proofs of the results stated in the paper.

2. Tree tensor networks

We consider functions \( f(x) = f(x_1, \ldots, x_d) \) defined on a product set \( \mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d \) and with values in \( \mathbb{R} \). Typically, \( \mathcal{X}_\nu \) is a subset of \( \mathbb{N} \) or \( \mathbb{R} \) but it could be a set of more general objects (vectors in \( \mathbb{R}^d_\nu \), sequences, functions, graphs...).

2.1. Tensor product feature space

For each \( \nu \in \{1, \ldots, d\} \), we introduce a finite-dimensional space \( V_\nu \) of functions defined on \( \mathcal{X}_\nu \). We let \( \{\phi^\nu_{i\nu} : i_{\nu} \in I^\nu\} \) be a basis of \( V_\nu \), with \( I^\nu = \{1, \ldots, N_\nu\} \). The functions \( \phi^\nu_{i\nu}(x_\nu) \) may be polynomials, splines, wavelets, kernel functions, or more general functions that extract \( N_\nu \) features from a given input \( x_\nu \in \mathcal{X}_\nu \). We let \( \phi^\nu : \mathcal{X}_\nu \rightarrow \mathbb{R}^{N_\nu} \) be the associated feature map defined by \( \phi^\nu(x_\nu) = (\phi^\nu_{1\nu}(x_\nu), \ldots, \phi^\nu_{N_\nu}(x_\nu))^T \in \mathbb{R}^{N_\nu} \). The functions \( \phi_i(x) = \phi^1_{i_1}(x_1) \cdots \phi^d_{i_d}(x_d) \), \( i \in I = I^1 \times \cdots \times I^d \), form a basis of the tensor product space \( V = V_1 \otimes \cdots \otimes V_d \). A function \( f \in V \) admits a representation

\[
 f(x) = \sum_{i \in I} a_i \phi_i(x) = \sum_{i_1=1}^{N_1} \cdots \sum_{i_d=1}^{N_d} a_{i_1, \ldots, i_d} \phi^1_{i_1}(x_1) \cdots \phi^d_{i_d}(x_d), \tag{2.1}
\]

where \( a \in \mathbb{R}^I = \mathbb{R}^{N_1 \times \cdots \times N_d} \) is an algebraic tensor (or multi-dimensional array) of size \( N_1 \times \cdots \times N_d \). The map \( \phi \) from \( \mathcal{X} \) to \( \mathbb{R}^I \) which associates to \( x \) the elementary tensor \( \phi(x) = \phi^1(x_1) \otimes \cdots \otimes \phi^d(x_d) \in \mathbb{R}^I \) defines a tensor product feature map.

Remark 2.1. In Section 6.1, we present a particular feature space based on tensorization of functions, that yields spaces \( V_\nu \) with a tensor product structure and an identification of \( f \) with a tensor of order higher than \( d \).

2.2. Tree-based ranks

For any \( \alpha \subset \{1, \ldots, d\} := D \), and \( x \in \mathcal{X} \), we denote by \( x_\alpha = (x_\nu)_{\nu \in \alpha} \in \mathcal{X}_\alpha \) the group of variables \( \alpha \) that take values in \( \mathcal{X}_\alpha = \times_{\nu \in \alpha} \mathcal{X}_\nu \). We let \( \alpha^c = D \setminus \alpha \).
Learning with tree tensor networks

Definition 2.2 (Ranks of multivariate functions and minimal subspaces). For a non-empty and strict subset $\alpha$ in $D$, the $\alpha$-rank of a function $f : \mathcal{X} \to \mathbb{R}$, denoted $\text{rank}_{\alpha}(f)$, is the minimal integer $r_{\alpha}$ such that

$$f(x) = \sum_{k=1}^{r_{\alpha}} g_k^\alpha(x_{\alpha}) h_k^\alpha(x_{\alpha^c}) \quad \text{(2.2)}$$

for some functions $g_k^\alpha : \mathcal{X}_\alpha \to \mathbb{R}$ and $h_k^\alpha : \mathcal{X}_{\alpha^c} \to \mathbb{R}$. The $\alpha$-dimensional subspace spanned by the functions $\{g_k^\alpha\}_{k=1}^{r_{\alpha}}$ is the $\alpha$-minimal subspace $U_{\alpha}^{\text{min}}(f)$ of $f$. For $\alpha = \emptyset$ or $\alpha = D$, we use the convention $\text{rank}_\emptyset(f) = 1$ and $\text{rank}_D(f) = 1$.

We let $T$ be a dimension partition tree over $D$, with root $D$ and leaves $\{\nu\}$, $1 \leq \nu \leq d$. For a node $\alpha \in T$, we denote by $S(\alpha)$ the set of children of $\alpha$. For any node $\alpha$, we have either $S(\alpha) = \emptyset$ (for leaf nodes) or $S(\alpha) \geq 2$ (for interior nodes). We denote by $\mathcal{L}(T)$ the set of leaves of $T$, and by $\mathcal{I}(T) = T \setminus \mathcal{L}(T)$ its interior nodes. For an interior node $\alpha \in \mathcal{I}(T)$, $S(\alpha)$ forms a partition of $\alpha$. The $T$-rank (or tree-based rank) of a function $f$ is the tuple $\text{rank}_T(f) = (\text{rank}_{\alpha}(f))_{\alpha \in T}$. The number of nodes of a dimension partition tree over $D$ is bounded as $|T| \leq 2d - 1$ (with equality for a binary tree).

Remark 2.3 (Vector-valued functions). The above definition and the subsequent notions can be easily extended to the case of vector-valued functions $f$ defined on $\mathcal{X}$ with values in $\mathbb{R}^s$ ($s \in \mathbb{N}$), by identifying $f$ with a function $\hat{f}(x_1, \ldots, x_d, i) = f_i(x_1, \ldots, x_d)$ of $d + 1$ variables. Most of the results of this paper then easily extends to this setting.

2.3. Tree tensor networks

Given a tuple $r = (r_{\alpha})_{\alpha \in T} \in \mathbb{N}^{|T|}$ we introduce the model class $M_T^r(V)$ of functions in $V$ with ranks bounded by $r$,

$$M_T^r(V) = \{ f \in V : \text{rank}_{\alpha}(f) \leq r_{\alpha}, \alpha \in T \}. $$

The set $M_T^r(V)$ is called a tree-based (or hierarchical) tensor format. A function $f \in M_T^r(V)$ admits a representation (2.2) for any $\alpha \in T$, with $\{g_k^\alpha\}_{k=1}^{r_{\alpha}}$ a basis of the minimal subspace $U_{\alpha}^{\text{min}}(f)$. From the definition of minimal subspaces, $f$ belongs to the tensor product space $\bigotimes_{\alpha \in \mathcal{S}(D)} U_{\alpha}^{\text{min}}(f)$, and therefore admits the representation (using tensor diagram notations\(^1\))

$$f(x) = \sum_{1 \leq k_{\alpha} \leq r_{\alpha} \text{ for } \alpha \in \mathcal{S}(D)} v_{1,(k_{\alpha})_{\alpha \in \mathcal{S}(D)}}^{D} \prod_{\alpha \in \mathcal{S}(D)} g_{k_{\alpha}}^\alpha(x_{\alpha}) = g_1^{x_1} \cdots g_{|\mathcal{S}(D)|}^{x_{|\mathcal{S}(D)|}}, \quad \text{(2.3)}$$

\(^1\)We use tensor diagram notations where each node represents a tensor and an edge connecting two nodes represents a contraction of two tensors over one of their modes.
where \( v^D \) is a tensor in \( \mathbb{R}^{\times_{\alpha \in S(T)} r_{\alpha}} \) and where \( g^\alpha(x_\alpha) = (g^\alpha_k(x_\alpha))_{1 \leq k \leq r_{\alpha}} \), with functions \( g^\alpha_k \in U^\alpha_\alpha(f) \subset V_\alpha \otimes_{\nu \in S(\alpha)} V_\nu \). From the nestedness property of minimal subspaces [18, Proposition 2], for any interior node \( \alpha \in \mathcal{I} \backslash \{D\} \), the functions \( g^\alpha_k \in \otimes_{\beta \in S(\alpha)} U^\beta_\beta(f) \) and therefore, they admit the representation

\[
g^\alpha_k(x_\alpha) = \sum_{1 \leq k \leq r_{\alpha}} v^\alpha_{k,\alpha, (k)} g^k_\alpha(x_\alpha) = \prod_{\beta \in S(\alpha)} g^\beta_\beta(x_\beta),
\]

(2.4)

where \( v^\alpha \in \mathbb{R}^{r_{\alpha} \times \times_{\beta \in S(\alpha)} r_{\beta}} \). For a leaf node \( \alpha \in \mathcal{L} \), the functions \( g^\alpha_k \in V_\alpha \) admit the representation

\[
g^\alpha_k(x_\alpha) = \sum_{i_\alpha \in I^\alpha} v^\alpha_{k,\alpha, i_\alpha} \phi^\alpha_{i_\alpha}(x_\alpha) = \prod_{\gamma \in \mathcal{L} \backslash \{D\}} g^\gamma_\gamma(x_\gamma),
\]

(2.5)

A function \( f \) in \( M^T_\alpha(V) \) therefore admits an explicit representation

\[
f(x) = \sum_{i_\alpha \in I^\alpha} \sum_{1 \leq k_\alpha \leq r_\alpha} \prod_{\gamma \in \mathcal{L} \backslash \{D\}} v^\gamma_{k_\gamma, (k_\alpha)} g^\gamma_\gamma(x_\gamma) \prod_{\gamma \in \mathcal{L} \backslash \{D\}} \phi^\gamma_{i_\gamma}(x_\gamma)
\]

(2.6)

where the set of parameters \( v = (v^\alpha)_{\alpha \in T} \) form a tree tensor network (see Figure 1b for a representation using tensor diagram notations). This tensor is defined by

\[
v^\alpha \in \mathbb{R}^{(1, \ldots, r_\alpha) \times I^\alpha := K^\alpha},
\]

with \( I^\alpha = \times_{\beta \in S(\alpha)} \{1, \ldots, r_\beta\} \) for \( \alpha \in \mathcal{I} \) or \( I^\alpha = \{1, \ldots, N_\alpha\} \) for \( \alpha \in \mathcal{L} \).

**Remark 2.4** (Tree tensor networks as compositional functions). A function associated with a tree tensor network \( v = (v^\alpha)_{\alpha \in T} \) admits a representation as a composition of multilinear functions, by identifying a tensor \( v^\alpha \in \mathbb{R}^{r_{\alpha} \times n_1 \times \ldots \times n_a} \) with a multilinear map.
Learning with tree tensor networks

from \( \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_2} \) to \( \mathbb{R}^r \). For example, for the dimension tree of Figure 1a, \( f \) admits the representation

\[
 f(x) = v^{1,\ldots,8} \left( v^{1,2,3}(v^1(x_1)), v^{2,3}(v^2(x_2)), v^3(x_3), v^4(x_4), v^5(x_5), v^{5,6,7}(v^5(x_5), v^6(x_6)), v^7(x_7), v^8(x_8) \right).
\]

For details, see Appendix C in supplementary materials.

A tensor network \( v = (v^\alpha)_{\alpha \in T} \) is said to be a sparse tensor network if the \( v^\alpha \) are sparse tensors. For \( \Lambda^\alpha \subset K^\alpha \), a tensor \( v^\alpha \) is said to be \( \Lambda^\alpha \)-sparse if \( v^\alpha_j = 0 \) for \( j \in K^\alpha \setminus \Lambda^\alpha \). For a given \( \Lambda = \times_{\alpha \in T} \Lambda^\alpha \), with \( K^\alpha \subset \Lambda^\alpha \), a tensor network \( v \) is said to be \( \Lambda \)-sparse if the \( v^\alpha \) are \( \Lambda^\alpha \)-sparse for all \( \alpha \in T \).

### 2.4. Parameter space and representation map

We introduce the product space of parameters

\[
P_{V,T,r} := \bigtimes_{\alpha \in T} P^\alpha, \quad P^\alpha := \mathbb{R}^{K^\alpha},
\]

and let \( \mathcal{R}_{V,T,r} \) be the map which associates to the tensor network \( v \in P_{V,T,r} \) the function \( f = \mathcal{R}_{V,T,r}(v) \) defined by (2.6), so that

\[
M_T(V) = \{ f = \mathcal{R}_{V,T,r}(v) : v \in P_{V,T,r} \}.
\]

From the representation (2.6), we obtain the following

**Lemma 2.5.** The representation map \( \mathcal{R}_{r,T,V} \) is a multilinear map from the product space \( P_{V,T,r} = \bigtimes_{\alpha \in T} P^\alpha \) to \( V \).

For \( \Lambda^\alpha \subset K^\alpha \), we denote by \( P^\alpha_{\Lambda^\alpha} \) the linear subspace of \( \Lambda^\alpha \)-sparse tensors in \( P^\alpha \). Then for \( \Lambda = \times_{\alpha \in T} \Lambda^\alpha \), we denote by \( P_{V,T,r,\Lambda} \subset P_{V,T,r} \) the set of \( \Lambda \)-sparse tensor networks and we introduce the corresponding model class

\[
M_{r,\Lambda}(V) = \{ f = \mathcal{R}_{V,T,r}(v) : v \in P_{V,T,r,\Lambda} \}.
\]

### 2.5. Complexity of a tensor network

When interpreting a tensor network \( v \in P_{V,T,r} = \bigtimes_{\alpha \in T} P^\alpha \) as a neural network, a classical measure of complexity is the number of neurons, which is the sum of ranks \( r^\alpha, \alpha \in T \). From an approximation or statistical perspective, a more natural measure of complexity is the number of parameters (or representation complexity), that is the dimension \( \sum_{\alpha \in T} \dim(P^\alpha) \) of the corresponding parameter space \( P_{V,T,r} \), or the number
of weights of the corresponding neural network. Then the representation complexity of \( v \) is

\[
C(T, r, V) := \sum_{\alpha \in T} |K^\alpha| = \sum_{\alpha \in I(T)} r_\alpha \prod_{\beta \in S(\alpha)} r_\beta + \sum_{\alpha \in L(T)} r_\alpha N_\alpha. \tag{2.7}
\]

For a sparse tensor network \( v \in P_{V, T, r, \Lambda} = \bigotimes_{\alpha \in T} P_{\Lambda^\alpha} \), a natural measure of complexity is given by

\[
C(T, r, V, \Lambda) = \sum_{\alpha \in T} |\Lambda^\alpha|, \tag{2.8}
\]

which only counts the number of non-zero parameters (or non-zero weights in the corresponding neural network). We note that \( C(T, r, V, \Lambda) \leq C(T, r, V) \). The different measures of complexity defined above lead to the definition of different approximation tools and corresponding approximation classes, see [1, 2, 3] for tensor networks, and [22] for similar results on ReLU or RePU neural networks.

3. Metric entropy of tree tensor networks

In this section, we provide an estimate of the metric entropy of the set of tree tensor networks (full or sparse) with normalized parameters. This is obtained by showing that tree tensor networks admit a Lipschitz parametrization.

We assume that the sets \( X_\nu \) are equipped with finite measures \( \mu_\nu \), for all \( \nu \in D = \{1, \ldots, d\} \), and the set \( X \) is equipped with the product measure \( \mu = \mu_1 \otimes \ldots \otimes \mu_d \). For \( 1 \leq p \leq \infty \), we consider the space \( L^p_\mu(X) \) of real-valued measurable functions defined on \( X \), with bounded norm \( \| \cdot \|_{p, \mu} \) defined by

\[
\| f \|_{p, \mu}^p = \int_X |f(x)|^p d\mu(x) \quad \text{for } 1 \leq p < \infty, \quad \text{or } \| f \|_{\infty, \mu} = \mu-\text{ess sup} |f|.
\]

If \( V_\nu \subset L^p_{\mu_\nu}(X_\nu) \) for all \( \nu \in D \), then \( V \subset L^p_\mu(X) \).

3.1. Normalized parametrization

A function \( f \in M^T_T(V) \) admits infinitely many equivalent parametrizations. From the multilinearity of the representation map \( R_{V,T,r} \) (see Lemma 2.5), it is clear that the model class \( M^T_T(V) \) is a cone, i.e. \( aM^T_T(V) \subset M^T_T(V) \) for any \( a \in \mathbb{R} \). Given some norms \( \| \cdot \|_{\alpha} \) on the spaces \( P^\alpha = \mathbb{R}^{K^\alpha} \), \( \alpha \in T \), and the corresponding product norm on \( P_{V,T,r} \) defined by

\[
\|(v^\alpha)_{\alpha \in T}\|_{P_{V,T,r}} = \max_{\alpha \in T} \|v^\alpha\|_{P^\alpha},
\]

we have

\[
M^T_T(V) = \{af : a \in \mathbb{R}, f \in M^T_T(V)\}_1,
\]
where $M^T_r(V)_1$ are elements of $M^T_r(V)$ with bounded parameters, defined by

$$
M^T_r(V)_1 = \{ f = R_{V,T,r}(v) : v \in \mathcal{P}_{V,T,r}, \|v\|_{\mathcal{P}_{V,T,r}} \leq 1 \}. \quad (3.1)
$$

The same normalization is used for defining the model class of sparse tensor networks

$$
M^T_{r,\Lambda}(V)_1 = M^T_{r,\Lambda}(V) \cap M^T_r(V)_1.
$$

### 3.2. Continuity of the parametrization

We study here the continuity of the representation map $R_{V,T,r}$ as a map from $\mathcal{P}_{V,T,r} = \times_{\alpha \in T} \mathcal{P}^\alpha$ to $V \subset L^p_\mu(X)$. From the multilinearity of $R_{V,T,r}$ (Lemma 2.5), we easily deduce the following property.

**Lemma 3.1.** Assuming $V \subset L^p_\mu(X)$, the multilinear map $R_{V,T,r}$ from $\mathcal{P}_{V,T,r}$ to $V \subset L^p_\mu(X)$ is continuous and such that for all $f = R_{V,T,r}(v_\alpha)_{\alpha \in T}$ in $M^T_r(V)$,

$$
\|f\|_{p,\mu} = \sup_{f = R_{V,T,r}(v_\alpha)_{\alpha \in T}} \prod_{\alpha \in T} \|v_\alpha\|_{\mathcal{P}^\alpha}.
$$

**Proof.** Denoting by $B(\mathcal{P}^\alpha)$ the unit ball of $\mathcal{P}^\alpha$ and by $B(\mathcal{P}_{V,T,r})$ the unit ball of $\mathcal{P}_{V,T,r}$. The set $M^T_r(V)_1$ defined by (3.1) is such that

$$
M^T_r(V)_1 = R_{V,T,r}(B(\mathcal{P}_{V,T,r})). \quad (3.3)
$$

We then deduce that the map $R_{V,T,r}$ is Lipschitz continuous on the set $M^T_r(V)_1$.

**Lemma 3.2.** Assuming $V \subset L^p_\mu(X)$, for all $f = R_{V,T,r}(v)$ and $\tilde{f} = R_{V,T,r}(\tilde{v})$ in $M^T_r(V)_1$,

$$
\|f - \tilde{f}\|_{p,\mu} \leq L_{p,\mu} \sum_{\alpha \in T} \|v^\alpha - \tilde{v}^\alpha\|_{\mathcal{P}^\alpha} \leq L_{p,\mu} |T| \|v - \tilde{v}\|_{\mathcal{P}_{V,T,r}}. \quad (3.4)
$$

**Proof.** Denoting by $\alpha_1, \ldots, \alpha_K$ the elements of $T$, we have

$$
f - \tilde{f} = \sum_{k=1}^K R_{V,T,r}(\tilde{v}^{\alpha_1}, \ldots, v^{\alpha_k} - \tilde{v}^{\alpha_k}, \ldots, v^{\alpha_K}).
$$

Then from Lemma 3.1, we obtain

$$
\|f - \tilde{f}\|_{p,\mu} \leq L_{p,\mu} \sum_{k=1}^K \|v^{\alpha_k} - \tilde{v}^{\alpha_k}\|_{\mathcal{P}^\alpha} \prod_{i < k} \|\tilde{v}^{\alpha_i}\|_{\mathcal{P}^\alpha} \prod_{i > k} \|v^{\alpha_i}\|_{\mathcal{P}^\alpha}, \quad (3.4)
$$

and we conclude by noting that $\|v^\alpha\|_{\mathcal{P}^\alpha} \leq 1$ and $\|\tilde{v}^\alpha\|_{\mathcal{P}^\alpha} \leq 1$ for all $\alpha \in T$. \qed
3.3. Metric entropy

The metric entropy $H(\epsilon, K, \| \cdot \|_X)$ of a compact subset $K$ of a normed vector space $(X, \| \cdot \|_X)$ is defined as

$$ H(\epsilon, K, \| \cdot \|_X) = \log N(\epsilon, K, \| \cdot \|_X), $$

with $N(\epsilon, K, \| \cdot \|_X)$ the covering number of $K$, which is the minimal number of balls of radius $\epsilon$ (for $\| \cdot \|_X$) necessary to cover $K$. We have the following result on the metric entropy of tensor networks with bounded parameters.

**Proposition 3.3.** Assume that $V \subset L^p_\rho(\mathcal{X})$, $1 \leq p \leq \infty$. The metric entropy of the model class

$$ M^T_r(V)_R = \{ af : a \in \mathbb{R}, |a| \leq R, f \in M^T_r(V) \} \quad (3.5) $$

in $L^p_\rho(\mathcal{X})$ is such that

$$ H(\epsilon, M^T_r(V)_R, \| \cdot \|_{p,\mu}) \leq C(T, r, V) \log(3\epsilon^{-1} R L_{p,\mu} |T|). $$

The metric entropy in $L^p_\rho(\mathcal{X})$ of the model class of $\Lambda$-sparse tensors

$$ M^T_{r,\Lambda}(V)_R = M^T_r(V)_R \cap M^T_{r,\Lambda}(V) \quad (3.6) $$

is such that

$$ H(\epsilon, M^T_{r,\Lambda}(V)_R, \| \cdot \|_{p,\mu}) \leq C(T, r, V, \Lambda) \log(3\epsilon^{-1} R L_{p,\mu} |T|). $$

**Proof.** The covering number of the unit ball $B(\mathcal{P}^\alpha)$ of the $|K^\alpha|$-dimensional space $\mathcal{P}^\alpha$ is such that $N(\epsilon, B(\mathcal{P}^\alpha), \| \cdot \|_{p,\alpha}) \leq (3\epsilon^{-1})^{|K^\alpha|}$. Then the unit ball $B(\mathcal{P}_{V,T,r})$ of the product space $\mathcal{P}_{V,T,r}$ equipped with the product topology has a covering number

$$ N(\epsilon, B(\mathcal{P}_{V,T,r}), \| \cdot \|_{p,\alpha}) \leq \prod_{\alpha \in T} N(\epsilon, B(\mathcal{P}^\alpha), \| \cdot \|_{p,\alpha}) \leq (3\epsilon^{-1})^{C(T,r,V)} $$

with $C(T, r, V) = \sum_{\alpha \in T} |K^\alpha|$. From the Lipschitz continuity of $\mathcal{R}_{V,T,r}$ on $M^T_r(V)_1$ (Lemma 3.2), we deduce that $N(\epsilon, M^T_r(V)_1, \| \cdot \|_{p,\mu}) \leq (3\epsilon^{-1} L_{p,\mu} |T|)^{C(T,r,V)}$, from which we deduce that $N(\epsilon, M^T_r(V)_R, \| \cdot \|_{p,\mu}) \leq (3\epsilon^{-1} R L_{p,\mu} |T|)^{C(T,r,V)}$, which ends the proof of the first statement. For sparse tensors, we first note that the unit ball $B(\mathcal{P}^\alpha_{X,\Lambda})$ of the $|\Lambda^\alpha|$-dimensional space $\mathcal{P}^\alpha_{X,\Lambda}$ is such that $N(\epsilon, B(\mathcal{P}^\alpha_{X,\Lambda}), \| \cdot \|_{p,\alpha}) \leq (3\epsilon^{-1})^{|\Lambda^\alpha|}$. Then a similar proof yields the desired upper bound with $C(T, r, V, \Lambda) = \sum_{\alpha \in T} |\Lambda^\alpha|$.

3.4. A particular choice of norms

Assume that $V \subset L^p_\rho(\mathcal{X})$. The continuity constant $L_{p,\mu}$ of the map $\mathcal{R}_{V,T,r}$ defined by (3.2) depends on $p, \mu$, the norms on parameter spaces $\mathcal{P}^\alpha$ and the chosen basis for $V$. 
Here we introduce a particular choice of norms and basis functions which allows to bound the continuity constant $L_{p,\mu}$. For any interior node $\alpha \in \mathcal{I}(T)$, we introduce a norm $\| \cdot \|_{\mathcal{P}_\alpha}$ over the space $\mathcal{P}_\alpha$ defined by

$$
\| v^\alpha \|_{\mathcal{P}_\alpha} = \max_{(z_\beta)_{\beta \in S(\alpha)} \in \mathbb{R}^{r_{\alpha}}} \frac{\| v^\alpha((z_\beta)_{\beta \in S(\alpha)}) \|_p}{\prod_{\beta \in S(\alpha)} \| z_\beta \|_p},
$$

where the tensor $v^\alpha \in \mathbb{R}^{r_{\alpha} \times (\times_{\beta \in S(\alpha)} r_\beta)}$ is identified with a multilinear map from $\times_{\beta \in S(\alpha)} \mathbb{R}^{r_\beta}$ to $\mathbb{R}^{r_{\alpha}}$, and where $\| \cdot \|_p$ refers to the vector $\ell^p$-norm (for more details, see Appendix C in supplementary materials). For a leaf node $\alpha \in \mathcal{L}(T)$, we introduce a norm $\| \cdot \|_{\mathcal{P}_\alpha}$ over the space $\mathcal{P}_\alpha$ defined by

$$
\| v^\alpha \|_{\mathcal{P}_\alpha} = \max_{z_\alpha \in \mathbb{R}^{N_\alpha}} \frac{\| v^\alpha(z_\alpha) \|_p}{\| z_\alpha \|_p},
$$

(3.7)

where the order-two tensor $v^\alpha \in \mathbb{R}^{N_\alpha \times r_{\alpha}}$ is identified with a linear map from $\mathbb{R}^{N_\alpha}$ to $\mathbb{R}^{r_{\alpha}}$. This corresponds to the matrix $p$-norm of $v^\alpha$. We assume that for any $\nu \in D$, the feature map $\phi^\nu : \mathcal{X}_\nu \rightarrow \mathbb{R}^{N_\nu}$ is such that $\| \phi^\nu \|_{p,\mu} = 1$. For $p = \infty$, that means that basis functions $\phi^\nu_t(x_\nu)$ have a unit norm in $L^\infty_{p,\mu}(\mathcal{X}_\nu)$. For $p < \infty$, that means that $\sum_{t=1}^{N_\nu} \| \phi^\nu_t \|_{p,\mu} = 1$, which can be obtained by rescaling basis functions so that $\| \phi^\nu_t \|_{p,\mu} = N_\nu^{-1/p}$.

**Proposition 3.4.** Assume $V \subset L^p_{\mu}(\mathcal{X})$, $1 \leq p \leq \infty$. With the above choice of norms and normalization of basis functions, the continuity constant $L_{p,\mu}$ defined by (3.2) is such that $L_{p,\mu} \leq 1$, and for all $1 \leq q \leq p$, $L_{q,\mu} \leq \mu(\mathcal{X})^{1/q-1/p} L_{p,\mu} \leq \mu(\mathcal{X})^{1/q-1/p}$.

**Proof.** See supplementary materials.

## 4. Risk bounds and model selection for tree tensor networks

Let $\mathcal{X}$ equipped with a finite measure $\mu$. In this section we analyze empirical risk minimization for contrasts computed over general families of functions associated to tree tensor networks built on approximation spaces in $L^\infty(\mathcal{X})$.

### 4.1. Risk bounds for tree tensor networks

We consider a model class $M$ of tensor networks with bounded parameters (with the norms defined in Section 3.4), with $M := M_T^T(V)_R$ for full tensor networks or $M_{T,\Lambda}(V)_R$ for $\Lambda$-sparse tensor networks. We consider here as fixed the approximation space $V$, the dimension tree $T$ and the ranks $r \in \mathbb{N}^{\{|T|\}}$, and also the sparsity pattern $\Lambda$ for sparse tensor networks.
networks. We assume that $V \subset L^\infty_{\mu}(\mathcal{X})$. We denote by $C_{M} = C(T, r, V)$ the representation complexity of $M$ defined by (2.7) for full tensor networks, or $C_{M} = C(T, r, V, \Lambda)$ the sparse representation complexity of $M$ defined by (2.8). We consider a risk

$$\mathcal{R}(f) = \mathbb{E}(\gamma(f, Z)),$$

where $Z$ is a random variable taking values in $\mathcal{Z}$ and where $\gamma : \mathbb{R}^{\mathcal{X}} \times \mathcal{Z} \rightarrow \mathbb{R}$ is some contrast function. The minimizer of the risk over measurable functions defined on $\mathcal{X}$ is the target function $f^\star$. For $f$ random (depending on the data), $\mathbb{E}(\gamma(f, Z))$ shall be understood as an expectation $\mathbb{E}_{Z}(\gamma(f, Z))$ w.r.t. $Z$ (conditional to the data). We introduce the excess risk

$$\mathcal{E}(f) = \mathcal{R}(f) - \mathcal{R}(f^\star).$$

Given the model class $M$, we denote by $f^M$ a minimizer over $M$ of the risk $\mathcal{R}$, and by $\hat{f}^M_n$ a minimizer over $M$ of the empirical risk

$$\hat{\mathcal{R}}_n(f) = \frac{1}{n} \sum_{i=1}^{n} \gamma(f, Z_i),$$

which is seen as an empirical process over $M$. To obtain bounds of the estimation error, it remains to quantify the fluctuations of the centered empirical process $\bar{\mathcal{R}}_n(f)$ defined by

$$\bar{\mathcal{R}}_n(f) = \hat{\mathcal{R}}_n(f) - \mathcal{R}(f) = \frac{1}{n} \sum_{i=1}^{n} \gamma(f, Z_i) - \mathbb{E}(\gamma(f, Z)).$$

**Assumption 4.1 (Bounded contrast).** Assume that $\gamma$ is uniformly bounded over $M \times \mathcal{Z}$, i.e.

$$|\gamma(f, Z)| \leq B$$

holds almost surely for all $f \in M$, with $B$ a constant independent of $f$.

**Assumption 4.2.** Assume that $\gamma(\cdot, Z)$ is Lipschitz continuous over $M \subset L^\infty_{\mu}(\mathcal{X})$, i.e.

$$|\gamma(f, Z) - \gamma(g, Z)| \leq \mathcal{L}\|f - g\|_{\infty, \mu}$$

holds almost surely for all $f, g \in M$, with $\mathcal{L}$ a constant independent of $f$ and $g$.

**Example A (Least-squares bounded regression).** We consider a random variable $Z = (X, Y)$, with $Y$ a random variable with values in $\mathbb{R}$, $X$ a $\mathcal{X}$-valued random variable with probability law $\mu$. We consider the least-squares contrast $\gamma(f, Z) = |Y - f(X)|^2$. The excess risk $\mathcal{E}(f) = \mathcal{R}(f) - \mathcal{R}(f^\star) = \|f - f^\star\|_{\infty, \mu}^2$ admits $f^\star(x) = \mathbb{E}(Y|X = x)$ as a minimizer. In the bounded regression setting, it is assumed that $|Y| \leq R$ almost surely.
For all $f \in M$, we have $\gamma(f, Z) \leq 2(|Y|^2 + \|f\|_\infty^2)$, so that $0 \leq \gamma(f, Z) \leq B$ almost surely, with $B = 4R^2$. Also, it holds almost surely

$$\left| \gamma(f, Z) - \gamma(g, Z) \right| = \left| (2Y - f(X) - g(X))(f(X) - g(X)) \right|$$

$$\leq (2|Y| + \|g\|_\infty + \|f\|_\infty)\|f - g\|_\infty.$$  

Then for all $f, g \in M$, $\left| \gamma(f, Z) - \gamma(g, Z) \right| \leq \mathcal{L}\|f - g\|_{\infty, \mu}$ with $\mathcal{L} = 4R$.

**Example B** ($L^2$ density estimation). For the problem of estimating the probability distribution of a random variable $X$, we consider $Z = X$. We consider the estimation of the probability law $\eta$ of $X$. Assuming that $\eta$ admits a density $f^*$ with respect to the measure $\mu$, and assuming $f^* \in L^2_\mu(\mathcal{X})$, we consider the contrast $\gamma(f, x) = \|f\|_2^2 - 2f(x)$, so that $\mathcal{E}(f) = \mathcal{R}(f) - \mathcal{R}(f^*) = \|f - f^*\|_2^2$ admits $f^*$ as a minimizer. We assume that $\mu$ is a finite measure on $\mathcal{X}$ and that $f^*$ is uniformly bounded by $R$. Then $\left| \gamma(f, X) \right| \leq B$ almost surely with $B = R(\mu(\mathcal{X})R + 2)$. Also, for all $f, g \in M$, we have almost surely

$$\left| \gamma(f, X) - \gamma(g, X) \right| = \left| \|f\|_2^2 - \|g\|_2^2 - 2(f(X) - g(X)) \right|$$

$$\leq \left| \int (f - g)(f + g)d\mu + 2\|f - g\|_\infty \right|$$

$$\leq (\|f + g\|_1 + 2)\|f - g\|_\infty$$

$$\leq \mathcal{L}\|f - g\|_{\infty, \mu}$$

with $\mathcal{L} = 2(\mu(\mathcal{X})R + 1)$.

**Proposition 4.3.** Under Assumptions 4.1 and 4.2, for any $t > 0$, with probability larger than $1 - \exp(-t)$,

$$\mathcal{E}(\hat{f}_n^M) \leq \mathcal{E}(f^M) + 8B\sqrt{C_M\sqrt{\frac{2\log(6LB^{-1}R|T|\sqrt{n})}{n}}} + 4B\sqrt{\frac{t}{2n}}.$$  

By integrating according to $t$, we obtain that

$$\mathbb{E}\mathcal{E}(f_n^M) \leq \mathcal{E}(f^M) + 8B\sqrt{C_M\sqrt{\frac{2\log(6LB^{-1}R|T|\sqrt{n})}{n}}} + 2B\sqrt{\frac{\pi}{n}}.$$  

This result is a standard application of the bounded difference inequality (see for instance Theorem 5.1 in [34]) applied to $\sup_{f \in M} |\overline{\mathcal{R}}_n(f)|$, together with a control on $\mathbb{E}\sup_{f \in M} |\overline{\mathcal{R}}_n(f)|$ with the metric entropy result of Proposition 3.3. The proof is given in the supplementary materials.

4.2. Model selection for tree tensor networks

We now consider a family of tensor networks $(M_m)_{m \in \mathcal{M}}$ indexed by a countable set $\mathcal{M}$. For full tensor networks, a model $M_m = M^{L_m}_m(V_m)_R$ is associated with a particular tree
\( T_m \), a rank \( r_m \), an approximation space \( V_m \), and a radius \( R \). For sparse tensor networks, a model \( M_m = M^{T_m}_{\Lambda_m}(V_m)_R \) has for additional parameter a sparsity pattern \( \Lambda_m \). We denote by \( C_m \) the number of parameters of the model \( M_m \), that is \( C_m = C(T_m, r_m, V_m) \) for full tensors, or \( C_m = C(T_m, r_m, V_m, \Lambda_m) \) for sparse tensors.

For some \( m \in \mathcal{M} \), we let \( f_m \) be a minimizer of the risk over \( M_m \),

\[
m_{\ast} = \arg \min_{f \in M_m} \mathcal{R}(f),
\]

and \( \hat{f}_m \) be a minimizer of the empirical risk over \( M_m \),

\[
\hat{f}_{\ast} = \arg \min_{f \in M_m} \hat{\mathcal{R}}_n(f).
\]

At this stage of the procedure, we have at hand a family of predictors \( \hat{f}_m \) and our goal is to provide a strategy for selecting a good predictor in the collection. We follow a standard strategy that corresponds to the so-called Vapnik’s structural minimization of the risk method (see for instance [34, Section 8.2]). Given some penalty function \( \text{pen} : \mathcal{M} \rightarrow \mathbb{R}^+ \), we define \( \tilde{m} \) as the minimizer over \( \mathcal{M} \) of the criterion

\[
\text{crit}(m) := \hat{\mathcal{R}}_n(\hat{f}_m) + \text{pen}(m),
\]

and we finally select the predictor \( \hat{f}_{\tilde{m}} \) according to the criterion (4.1). This procedure is classical in non parametric statistics and similar model selection approaches can be found in [41, 24, 10].

For a suitable choice of penalty which takes into account both the complexity of the models and the richness of the model collection, we provide a risk bound for the selected predictor. Let

\[
N_c := N_c(\mathcal{M}) = |\{ m \in \mathcal{M} : C_m = c \}|
\]

be the number of models with complexity \( c \) in the collection. The following result corresponds to the general Theorem 8.1 in [34] applied to our framework.

**Theorem 4.4.** Let \( \bar{w} > 0 \). Under Assumptions 4.1 and 4.2, if the penalty is such that

\[
\text{pen}(m) \geq \lambda_m \sqrt{\frac{C_m}{n}} + 2B \sqrt{\bar{w}C_m + \log(N_{C_m})},
\]

with

\[
\lambda_m = 4B \sqrt{2 \log(6LB^{-1}RT_m|\sqrt{n})},
\]

then the estimator \( \hat{f}_{\tilde{m}} \) selected according to the criterion (4.1) satisfies the following risk bound

\[
\mathbb{E}(\mathcal{E}(\hat{f}_{\tilde{m}})) \leq \inf_{m \in \mathcal{M}} \{ \mathcal{E}(f_m) + \text{pen}(m) \} + \frac{B}{\exp(\bar{w}) - 1} \sqrt{\frac{n}{2n}}.
\]

**Proof.** The proof of Theorem 4.4 is given in the supplementary materials, it is a direct adaptation of the proof of Theorem 8.1 in [34].
4.3. Collections of models and their richness

Here we present and analyze the richness of different collections of tensor networks $(M_m)_{m \in \mathcal{M}}$, where each model has a particular feature space $V_m$, a tree $T_m$, a tuple of ranks $r_m$. These collections of models depend on whether the feature space and the tree are considered as fixed. More precisely, we consider the following collections of models $(M_m = M_{r_m}^{T_m}(V_m))_{m \in \mathcal{M}}$ with $\mathcal{M}$ corresponding to one of the following collections:

- $\mathcal{M}_{V,T}$: fixed feature space $V_m = V$, fixed tree $T_m = T$, variable ranks $r_m$,
- $\mathcal{M}_T$: variable feature space $V_m$, fixed tree $T$, variable ranks $r_m$,
- $\mathcal{M}_*$: variable feature space $V_m$, variable tree $T_m$, variable ranks $r_m$.

For variable feature spaces, we classically consider that $V_m := V_{N_m}$ with $N_m \in \mathbb{N}^d$ and for any $N \in \mathbb{N}^d$, $V_N = V_{1,N} \otimes \ldots \otimes V_{d,N}$, where $(V_{\nu,N_\nu})_{\nu \in \mathbb{N}}$ is a sequence of subspaces of univariate functions, with $N_\nu = \dim(V_{\nu,N_\nu})$. For variable trees, we consider trees in the family of trees with arity $a$ (or $a$-ary trees), the case $a = 2$ corresponding to (full) binary trees. The next result provides upper bounds of the complexity of the above defined families of tensor networks.

**Proposition 4.5** (Collections of full tensor networks). Consider a family of full tensor networks $(M_m = M_{r_m}^{T_m}(V_m))_{m \in \mathcal{M}}$ with $\mathcal{M}$ equal to $\mathcal{M}_{V,T}$, $\mathcal{M}_T$ or $\mathcal{M}_*$. For any tree $T$ and any feature space $V$, $\mathcal{N}_c(\mathcal{M}_{V,T}) \leq \mathcal{N}_c(\mathcal{M}_T) \leq \mathcal{N}_c(\mathcal{M}_*)$, and

$$\log(\mathcal{N}_c(\mathcal{M}_*)) \leq 2a(c + d \log(c)),$$

with $a$ the arity of the considered trees.

**Proof.** See supplementary materials.

When exploiting sparsity, we consider models $M_m = M_{r_m}^{T_m}(V_m)$ depending on an additional sparsity pattern $\Lambda_m$. For variable feature spaces $V_m = V_{N_m}$, we consider models $m$ such that $N_m \in \mathbb{N}^d$ satisfies

$$N_m \leq g(C_m), \quad \text{(4.4)}$$

with $g$ some increasing function of the complexity $C_m$ of the model $m$. This is a reasonable assumption from a practical point of view, where for a given complexity, we avoid the exploration of infinitely many features. We use the same notations $\mathcal{M}_{V,T}$, $\mathcal{M}_T$ and $\mathcal{M}_*$ for the corresponding families of models, with $\Lambda_m$ considered as an additional free variable. The complexities of these collections of sparse tensor networks are higher than the corresponding complexities for full tensor networks, but only up to logarithmic terms, as shown in the next result.

**Proposition 4.6** (Collections of sparse tensor networks). Consider a family of sparse tensor networks $(M_m = M_{r_m}^{T_m}(V_{N_m}))_{m \in \mathcal{M}}$ with $\mathcal{M}$ equal to $\mathcal{M}_{V,T}$, $\mathcal{M}_T$ or $\mathcal{M}_*$, with
variable sparsity patterns $\Lambda_m$ and $N_m \leq g(C_m)$. For any tree $T$ and any feature space $V$, $N_c(M_{V,T}) \leq N_c(M_T) \leq N_c(M_*)$, and

$$\log(N_c(M_*)) \leq 5ac\log(c) + 2c\log(g(c)).$$

If we further assume that $\log(g(c)) \leq \delta \log(c)$ for some $\delta > 0$, then

$$\log(N_c(M_*)) \leq (5a + 2\delta)c\log(c).$$

Proof. See supplementary materials.

Together with Proposition 4.5 (or Proposition 4.6), Theorem 4.4 provides a strong justification for using a penalty proportional to $\sqrt{C_m/n}$. However, it is known that the Vapnik’s structural minimization of the risk may lead to suboptimal rates of convergence. For instance, in the bounded regression setting, it is known that a penalty proportional to the VapnikChervonenkis dimension (typically in $O(C_m/n)$) leads to minimax rates of convergence in various setting (see for instance Chapter 12 in [24]) whereas Vapnik’s structural minimization of the risk (typically with penalty in $O(\sqrt{C_m/n})$) is too pessimistic to provide fast rates of convergence.

In the case of bounded least squares contrasts, we give in Section 5 improved risk bounds and next we show in Section 6 that our model selection strategy is (near to) adaptive minimax in several frameworks.

5. Oracle inequality for least squares inference with tree tensor networks

In this section, we provide an improved excess risk bound in the specific case of least squares contrasts. Our results come from Talagrand inequalities and generic chaining bounds; we follow the presentation given in the monograph [32]. The excess risk bound given below strongly relies on the link between the excess risk and the variance of the excess loss, as explained in Chapter 5 of [32] or Chapter 8 in [34]. We then derive an improved model selection result for least squares inference by following the approach presented in Sections 8.3 and 8.4 of [34] or in Section 6.3 of [32].

Let $\gamma$ be either the least squares contrast in the bounded regression setting (as described in Example A), or the least squares contrast for density estimation (as described in Example B).

5.1. Improved risk bounds for least squares contrasts

We first consider as model class a tree tensor network $M = M^T_r(V)_R$ or $M = M^T_{r,\Lambda}(V)_R$ (respectively full or $\Lambda$-sparse) with bounded parameters and it assumed that the feature tensor space $V \subset L^\infty_\mu(X)$ where $\mu$ is the distribution of the random variable $X$ in the regression setting (see Example A) or the reference measure for density estimation (see Example B).
Proposition 5.1. Under Assumptions 4.1 and 4.2, there exists an absolute constant \( A \) and a constant \( \kappa \) such that for any \( \varepsilon \in (0,1] \) and any \( t > 0 \), with probability at least \( 1 - A \exp(-t) \), it holds
\[
E(\hat{f}^M_n) \leq (1 + \varepsilon) E(f^M) + \frac{K_2 R^2}{n} \left[ \frac{a_T C_M}{\varepsilon^2} \log^+ \left( \frac{n \varepsilon^2}{a_T C_M} \right) + \frac{t}{\varepsilon} \right]
\]
(5.1)
where \( a_T = 1 + \log^+ \left( \frac{3|T|^4}{4e} \right) \), and \( \kappa \) depends linearly on \( \mu(\mathcal{X})^2 \). Then by integrating according to \( t \), we obtain that for any \( \varepsilon \in (0,1] \),
\[
\mathbb{E} E(\hat{f}^M_n) \leq (1 + \varepsilon) E(f^M) + \frac{K_2 R^2}{n} \left[ \frac{a_T C_M}{\varepsilon^2} \log^+ \left( \frac{n \varepsilon^2}{a_T C_M} \right) + \frac{A}{\varepsilon} \right].
\]

Proof. The proof of the proposition is given in the supplementary materials.

Note that the term \( a_T \) is upper bounded by a term of the order of \( \log(d) \) because \( |T| \leq 2d \). Thus the constants in the risk bound (5.1) do not explode with the dimension \( d \) in regression. Note however that in density estimation, the constant \( \kappa \) depends linearly on the mass \( \mu(\mathcal{X}) \) of the reference measure, which may grow exponentially with \( d \).

5.2. Oracle inequality

As in Section 4.2, we now consider a family of tensor networks \((M_m)_{m \in M}\) indexed by a countable set \( M \), with either \( M_m = M^{T_m}_{r_m}(V_m)_{R_m} \) for full tensor networks, or \( M_m = M^{T_m}_{r_m,\Lambda_m}(V_m)_{R_m} \) for sparse tensor networks. We consider features spaces \( V_m \subset L^\infty_\mu(X) \) with \( X \) equipped with a finite measure \( \mu \). As before, \( N_c(M) \) denotes the number of models with complexity \( c \) in the collection \( M \) (see Section 4.3).

Theorem 5.2. Let \( \bar{w} > 0 \). Under Assumptions 4.1 and 4.2, there exists numerical constants \( K_1 \) and \( K_2 \) and \( K_3 \) such that if the penalty satisfies
\[
\text{pen}(m) = K_1 R^2 \left[ \frac{b_m C_m}{n \varepsilon^2} \log^+ \left( \frac{n \varepsilon^2}{b_m C_m} \right) + \frac{\bar{w} C_m + \log(N_{C_m})}{n \varepsilon} \right]
\]
with \( b_m = 1 + \log^+ \left( \frac{3|T_m|^4}{4e} \right) \), then the estimator \( \hat{f}_m \) selected according to the penalized criterion (4.1) satisfies the following oracle inequality
\[
\mathbb{E} E(\hat{f}_m) \leq \frac{1 + \varepsilon}{1 - \varepsilon} \inf_{m \in M} \{ E(f_m) + K_2 \text{pen}(m) \} + \frac{K_3 R^2}{\exp(\bar{w}) - 1 \varepsilon} \frac{1 + \varepsilon}{1 - \varepsilon} \frac{1}{n}.
\]
(5.2)

\( ^2 \)With \( \mu(\mathcal{X}) = 1 \) for regression.
Proof. The proof, adapted from Theorem 6.5 in [32], is given in the supplementary materials.

For collections of models $\mathcal{M}$ such that\footnote{$A \sim B$ means that there exist constants $c$ and $C$ such that $cB \leq A \leq CB$}
\[ \log(N_{C_m}(\mathcal{M})) \sim C_m \log(C_m) \delta \]
for some $\delta \geq 1$, this theorem provides an improved oracle inequality bound\footnote{$A \lesssim B$ means that there exists a constant $C$ such that $A \leq CB$}
\[ \mathbb{E}\mathcal{E}(\hat{f}_m) \lesssim \inf_{m \in \mathcal{M}} \mathcal{E}(f_m) + \frac{C_m}{n} \log(n) \log(C_m) \delta, \quad (5.3) \]
with a penalty in $\frac{C_m}{n}$, up to logarithmic terms.

From this model selection result, we derive in Section 6 adaptive (near to) optimal rates of convergence (in the minimax sense) for several smoothness classes. In Section A.1 we explain how to calibrate the penalty in practice using the slope heuristics method.

6. Least-squares inference and minimax adaptivity for smoothness classes

We consider here bounded least-squares inference for target functions $f^*$ belonging to classical smoothness spaces including Sobolev or Besov spaces (with isotropic, anisotropic or mixed dominating smoothness), or spaces of analytic functions. We consider functions defined on the hypercube $[0, 1]^d$ equipped with the uniform measure $\mu$. For clarity, we let $L_p := L_p^\mu([0, 1]^d)$.

A classical approach is to consider tensor networks with feature tensor spaces $V_m$ that are adapted to the smoothness of the function (e.g. tensorized splines or wavelets for Besov smoothness, or tensorized polynomials for analytic functions). Here, we use an alternative and powerful approach based on tensorization of functions, which can be interpreted as a particular definition of features. It does not require to adapt the tool to the regularity of the function. This approach is described in Section 6.1 and Section 6.2 (for more details see [1, 3]). Then in Section 6.3, we show that our model selection strategy with this tool is minimax adaptive to a wide range of smoothness classes.

6.1. Feature space based on tensorization of functions at fixed resolution

For any integers $b, L \in \mathbb{N}$ with $b \geq 2$, we introduce an uniform partition of the interval $[0, 1)$ into $b^L$ intervals of equal length $b^{-L}$. Any $x \in [0, 1)$ can be written as
\[ x = \sum_{k=1}^{L} i_k b^{-k} + b^{-L} \bar{x} := t_{b,L}(i_1, \ldots, i_L, \bar{x}), \]

\[ A \sim B \] means that there exist constants $c$ and $C$ such that $cB \leq A \leq CB$\footnote{$A \lessapprox B$ means that there exists a constant $C$ such that $A \leq CB$}
where \((i_1, \ldots, i_L) \in \{0, \ldots, b-1\}^L\) is the representation in base \(b\) of the integer \(i\) such that \(x \in [b^{-L}i, b^{-L}(i+1))\), and \(\bar{x} \in [0, 1)\). The integer \(L\) is called the resolution. The map \(t_{b,L}\) is a bijection from \([0, \ldots, b-1]^d \times [0, 1)\) to \([0, 1)^L\) with inverse \(t^{-1}_{b,L}(x) = (i_1, \ldots, i_L, \bar{x})\) such that 
\[
i_k = [b^k x \mod b], \quad \bar{x} = b^L x - [b^L x].
\]

A function \(f(x)\) defined on \([0, 1)\) can then be linearly identified with a \((L+1)\)-variate function \(f(i_1, \ldots, i_L, \bar{x})\) defined on \([0, \ldots, b-1]^L \times [0, 1)\). The map \(T_{b,L}\) which associates to a function \(f\) the multivariate function \(f \circ t^{-1}_{b,L}\) defined on \([0, \ldots, b-1]^L \times [0, 1)\) is called the tensorization map.

For multivariate functions \(f(x_1, \ldots, x_d)\) defined on the hypercube \([0, 1)^d\), we proceed in a similar way for each dimension. Each variable \(x_\nu\) is identified with a tuple \((i_{\nu,1}, \ldots, i_{\nu,d}, \bar{x}_\nu) = t_{b,L}(x_\nu)\), and \(f\) is linearly identified with a \(d(L+1)\)-variate function \(f(i_{\nu,1}, \ldots, i_{\nu,d}, i_1, \ldots, i_L, \bar{x}_1, \ldots, \bar{x}_d)\) defined on \([0, \ldots, b-1]^{Ld} \times [0, 1)^d\).

For any \(1 \leq p \leq \infty\), the tensorization map \(T_{b,L}\) which associates to a \(d\)-variate function \(f\) the tensor \(f\) of order \((L+1)d\) is a linear isometry from \(L^p([0, 1)^d)\) to the tensor Banach space \((\mathbb{R}^b)^{\otimes Ld} \otimes L^p([0, 1)^d) = L^p_\mu([0, \ldots, b-1]^{Ld} \times [0, 1)^d)\) equipped with the uniform measure \(\mu\) over \([0, \ldots, b-1]^{Ld} \times [0, 1)^d\) [3, Theorem 2.2].

To define an approximation tool, we then introduce a finite-dimensional tensor space 
\[
V_L = (\mathbb{R}^b)^{\otimes L} \otimes (\mathbb{P}_k)^{\otimes d}
\]
where \(\mathbb{P}_k\) is the space of univariate polynomials of degree less than \(k\). To a tensor \(f \in V_L\) corresponds a function \(\tilde{f} = T_{b,L}^{-1} f \in L^\infty([0, 1)^d)\) which is a spline of degree \(k\) on the uniform partition of \([0, 1)\).

\section{6.2. Tensor networks with variable resolution: complexity and approximation classes}

Here we consider tensor networks over the tensor space \(V_L\), either \(M^T_r(V_L)\) for full tensor networks or \(M^{s,\Lambda}_{r,n}(V_L)\) for sparse tensor networks, where \(T\) is a dimension tree over \([1, \ldots, d(L+1)]\), \(r \in \mathbb{N}^{\lceil T \rceil}\), and \(\Lambda\) some sparsity pattern. This defines a subset of \(d\)-variate functions through the map \(T_{b,L}^{-1}\). For a linear tree
\[
T = T_L := \{\{1\}, \ldots, \{d(L+1)\}, \{1, 2\}, \{1, 2, 3\}, \ldots, \{1, \ldots, d(L+1)\}\},
\]
the tensor network corresponds to a tensor train (TT) format.

\textbf{Remark 6.1.} For the approximation of functions from classical smoothness classes, and when working with a fixed tree, this choice of tree is rather natural. Each interior node in \(T_L\) is related to a splitting of variables into a group of low-resolution variables and high-resolution variables (see discussions in [1, 3] on the impact of the tree).
6.2.1. Collections of tensor networks and their richness.

We consider as an approximation tool a collection of tensor networks with variable resolutions and variable ranks with a tensor train format. More precisely, we define a collection of models \( (M_m)_{m \in M} \) in \( L^\infty([0,1]^d) \) defined by

\[
M_m = T_{b,L_m}^{-1} M_{r_m,L_m}^{T_{L_m}} (V_{L_m})_R \quad \text{or} \quad M_m = T_{b,L_m}^{-1} M_{r_m,L_m}^{T_{L_m}} (V_{L_m})_R,
\]

with variable resolutions \( L_m \in \mathbb{N} \), linear trees \( T_{L_m} \) and variable ranks \( r_m \in \mathbb{N} \).

Remark 6.2. Note that for a particular resolution \( L_m \), we consider here a single tree \( T_{L_m} \). This is sufficient for obtaining our minimax results for classical smoothness classes in Section 6.3. Working with variable trees may be relevant for highly structured functions or functions beyond classical smoothness classes. Our tree selection procedure should be able to recover a near-optimal tree, that is relevant for applications where there is no a priori for the selection of a good tree.

Note that since \( T_{b,d} \) is a linear isometry from \( L^p_\mu \) to \( L^p_\mu \), the metric entropy \( H(\epsilon, M_m, \| \cdot \|_{p,\mu}) \) of \( M_m \) is equal to the metric entropy of the corresponding tensor network in \( L^p_\mu \).

For a model \( m \) with complexity \( c \), we clearly have \( L_m \leq c \). Then the number \( N_c(M) \) of models with complexity \( c \) is such that

\[
N_c(M) = \sum_{L=0}^\infty N_c(M_{T_L}) = \sum_{L=0}^c N_c(M_{T_L}),
\]

with \( M_{T_L} \) the collection of tensor networks with fixed tree \( T_L \), fixed feature space \( V_L \), and variable ranks. We deduce from Proposition 4.5 (with \( d \) replaced by \( (L+1)d \leq c \)) and Proposition 4.6 (with a constant function \( g(c) = \max\{b,k+1\} \)) that

\[
N_c(M) \lesssim c \log(c), \quad (6.1)
\]

for both full and sparse tensor networks.

Given a collection of tensor networks \( (M_m)_{m \in M} \) introduced above (either full or sparse), we define an approximation tool \( \Phi = (\Phi_c)_{c \in \mathbb{N}} \), where the set \( \Phi_c \) is the union of models with complexity less than \( c \), i.e.

\[
\Phi_c = \bigcup_{m \in M, C_m \leq c} M_m.
\]

The approximation tool \( \Phi \) is respectively denoted by \( \Phi^F = (\Phi^F_c)_{c \in \mathbb{N}} \) and \( \Phi^S = (\Phi^S_c)_{c \in \mathbb{N}} \) for full and sparse tensor networks.

6.2.2. Approximation classes

The best approximation error of \( f^* \) in \( L^2 \) by a tensor network with complexity less than \( c \) is

\[
E(f^*, \Phi_c)_{L^2} = \inf_{f \in \Phi_c} \mathcal{E}(f)^{1/2} = \inf_{f \in \Phi_c} \| f - f^* \|_{2,\mu}.
\]

imsart-bj ver. 2014/10/16 file: model_selection_tensors_bernoulli_second_revision.tex date: May 13, 2021
Then given a growth function $\gamma : \mathbb{N} \rightarrow \mathbb{N}$, an approximation class for tensor networks can be defined as the set of functions

$$\mathcal{A}_\infty(\gamma, \Phi, L^2) = \{ f : \sup_{c \geq 1} \gamma(c) E(f, \Phi_c)_{L^2} < \infty \},$$

which corresponds to functions that can be approximated with tree tensor networks with an error $E(f^*, \Phi_c)$ in $O(\gamma(c)^{-1})$.

To polynomial growth functions $\gamma(c) = c^\alpha$ ($\alpha > 0$) correspond approximation classes

$$\mathcal{A}_\alpha^\infty := \mathcal{A}_\infty^\alpha(\Phi, L^2) = \{ f : \sup_{c \geq 1} c^\alpha E(f, \Phi_c)_{L^2} < \infty \},$$

containing functions that can be approximated by tensor networks with algebraic convergence such that $E(f^*, \Phi_c) \lesssim c^{-\alpha}$. In [1, 3], it is proved that the sets $\mathcal{A}_\infty^\alpha$ are quasi-Banach spaces, equipped with the quasi-norm $\|f\|_{\mathcal{A}_\infty^\alpha} = \|f\|_{L^2} + \|f\|_{\mathcal{A}_\infty^\alpha}$ with $\|f\|_{\mathcal{A}_\infty^\alpha} = \sup_{c \geq 1} c^\alpha E(f, \Phi_c)_{L^2}$. A whole range of quasi-Banach spaces $\mathcal{A}_\beta^q$ can be defined by interpolation between $L^2$ and a space $\mathcal{A}_\alpha^\infty$, with $\mathcal{A}_\beta^q = (L^2, \mathcal{A}_\alpha^\infty)_{\beta/\alpha, q}$, $0 < \beta < \alpha$, $0 < q \leq \infty$. The spaces $\mathcal{A}_\alpha^\infty$ are included in $\mathcal{A}_\infty^\alpha$ and correspond to a slightly stronger convergence of approximation error.

The approximation classes associated with full and sparse tensor networks (associated with two different notions of complexity) are respectively denoted by

$$\mathcal{F}_q^\alpha = \mathcal{A}_q^\alpha(\Phi^F, L^2) \quad \text{and} \quad \mathcal{S}_q^\alpha = \mathcal{A}_q^\alpha(\Phi^S, L^2).$$

For any $0 < q \leq \infty$, we have the following continuous embeddings [3, Theorem 4.12]

$$\mathcal{F}_q^\alpha \hookrightarrow \mathcal{S}_q^\alpha \hookrightarrow \mathcal{F}_q^{\alpha/2}. \quad (6.2)$$

That means that if full tensor networks achieve an approximation rate as $O(c^{-\alpha})$ then sparse tensor networks achieve at least the same approximation rate. However, if sparse tensor networks achieve an approximation rate as $O(c^{-\alpha})$, then full tensor networks achieve at least an approximation rate as $O(c^{-\alpha/2})$, i.e. with a possible deterioration of the rate by a factor 2.

**Remark 6.3.** We recall that the results of this section are valid for a collection of models where for a given resolution $L$, we consider a single tree $T_L$. When considering variable trees for a fixed resolution, we obtain much larger approximation classes. However, these are highly nonlinear classes and their properties have not been studied yet.

### 6.3. Rates for smoothness classes

Here we show that (near to) minimax rates can be achieved by tensor networks with our model selection strategy for a wide range of smoothness classes encompassing isotropic Besov spaces, anisotropic Besov spaces, Besov spaces with mixed dominating smoothness and spaces of analytic functions.
For that, we rely on the oracle inequality from Section 5.1, the estimates of the complexity of collections of tensor networks from Section 6.2, and approximation results from [2, 3].

Next we denote by \( \hat{f}_m^F \) and \( \hat{f}_m^S \) the estimators obtained with our model selection strategy using full tensor networks or sparse tensor networks respectively.

In this section, we work under the assumptions of Theorem 5.2.

6.3.1. Besov spaces with isotropic smoothness

We let \( B^s_q(L^p) \) denote the Besov space of functions with regularity order \( s > 0 \), primary parameter \( p \) and secondary parameter \( q \) (see [3, 14] for a definition and characterization). The parameter \( p \) is related to the norm with which the regularity is measured.

For \( s < 1 \) and \( q = \infty \), \( B^s_\infty(L^p) \) corresponds to the space \( \text{Lip}(s,L^p) \). For \( p = q \) and non-integer \( s > 0 \), \( B^s_q(L^p) \) corresponds to the (fractional) Sobolev space \( W^{s,p} \). For the special case \( p = 2 \), \( B^s_2(L^2) \) is equal to the Sobolev space \( W^{s,2} = H^s \) for any \( s > 0 \). For \( s > d(1/\tau - 1/p)_+ \), it holds that \( B^s_q(L^\tau) \hookrightarrow L^p \).

It is known that the minimax rate for functions \( f^* \in B^s_q(L^p) \) is lower bounded by \( n^{-\frac{2s}{2s+d}} \) (see e.g. [15, 19]).

**Besov spaces \( B^s_q(L^p) \) for \( p \geq 2 \).** We first consider Besov spaces \( B^s_q(L^p) \) with smoothness measured in \( L^2 \) norm or stronger norm.

**Theorem 6.4** (Minimax rates for Besov spaces \( B^s_q(L^p) \) for \( p \geq 2 \)). Assume the target function \( f^* \in B^s_q(L^p) \) with \( s > 0 \), \( 2 \leq p \leq \infty \) and \( 0 < q \leq \infty \). Then for sufficiently large \( n \),

\[
\mathbb{E} \| \hat{f}_m^F - f^* \|_{2,\mu}^2 \lesssim n^{-\frac{2s}{2s+d}} \log(n)^{\frac{d}{2s+d}}
\]

with \( \hat{s} = s \) if \( k \geq s - 1/2 \) or an arbitrary \( \hat{s} < s \) if \( k < s - 1/2 \).

**Proof.** See supplementary materials.

The above theorem implies that our model selection procedure with full tensor networks achieves minimax rates (up to logarithmic term) for the whole range of Besov spaces \( B^s_q(L^p) \), \( p \geq 2 \). It is thus minimax adaptive to the regularity over these Besov spaces, i.e. it achieves minimax rates without the need to adapt the approximation tool to the regularity of the target function. Note that minimax rates for \( B^s_q(L^p) \), \( p \geq 2 \), are also achieved with linear approximation tools such as splines, wavelets or kernel methods, but obtaining minimax adaptivity requires a suitable strategy for the selection of a particular family of splines, wavelets or kernels. Here, tensor networks are associated with spline functions of a fixed degree \( k \), and minimax adaptivity is obtained for any fixed value of \( k \), including \( k = 0 \). This is made possible by allowing models with high resolution (corresponding to deep tensor networks).
Besov spaces $B^s_\tau(L^r)$ for $\tau < 2$. Now we consider the case of Besov spaces $B^s_\tau(L^r)$ with a regularity measured in a weaker $L^r$-norm, $\tau < 2$. These are spaces of functions with "inhomogeneous smoothness" that can be only well captured by nonlinear approximation tools. We consider spaces $B^s_\tau(L^r)$ with $1/2 < 1/\tau < s/d + 1/2$. In the usual $(1/\tau, s)$ DeVore diagram of smoothness spaces, this corresponds to Besov spaces strictly above the critical line characterized by $s = d(1/\tau - 1/2)$. Besov spaces strictly above this line ($s > d(1/\tau - 1/2)$) are compactly embedded in $L^2$, while Besov spaces strictly below this line ($s < d(1/\tau - 1/2)$) are not embedded in $L^2$.

**Theorem 6.5** (Minimax rates for Besov spaces $B^s_\tau(L^r)$). Assume the target function $f^* \in B^s_\tau(L^r)$ with $s > 0$, $1/2 < 1/\tau < s/d + 1/2$ and $0 < q \leq \tau$. Then for sufficiently large $n$, the estimators using full or sparse tensor networks respectively satisfy

$$
\mathbb{E}\|\hat{f}_n^s - f^*\|_2^2 \lesssim n^{-\frac{2s}{2s+1}} \log(n)^{\frac{2s}{2s+1}}
$$

and

$$
\mathbb{E}\|\hat{f}_n^s - f^*\|_{2,\mu}^2 \lesssim n^{-\frac{2s}{2s+1}} \log(n)^{\frac{2s}{2s+1}}
$$

with $s$ if $k \geq s - 1/2$ or an arbitrary $\hat{s} < s$ if $k < s - 1/2$.

**Proof.** See supplementary materials.

For such spaces $B^s_\tau(L^r)$ above the critical line and $\tau < 2$, it is known that optimal linear estimators do not achieve the optimal rate. For $d = 1$, optimal linear estimators achieve a rate in $n^{-\frac{2s}{2s+1}}$, which is larger than the minimax rate $n^{-\frac{2s}{2s+1}}$. Only nonlinear methods of estimation are able to achieve the minimax rate [16]. The above result shows that our model selection strategy with sparse tensor networks achieves minimax rates or rates arbitrarily close to minimax (up to a logarithmic term) for the whole range of spaces $B^s_\tau(L^r)$, without requiring to adapt the tool to the regularity. Note that the estimation using full tensor networks presents a slightly deteriorated rate. In this nonlinear estimation setting, exploiting sparsity of the tensor network is useful to obtain an optimal performance. Note that the chosen polynomial degree $k$ has only a little impact on the obtained results. If this degree is adapted to the regularity ($k \geq s - 1/2$), the minimax rate is achieved (up to logarithmic term) but any degree $k$ (including $k = 0$) allows to achieve a rate arbitrarily close to optimal.

**6.3.2. Besov spaces with mixed dominating smoothness**

Here we consider Besov spaces $MB^s_q(L^p)$ with mixed dominating smoothness (see [3, 27, 26] for a definition and characterization). For $p = q = 2$, $MB^s_q(L^2)$ corresponds to the mixed Sobolev spaces $H^{s,\text{mix}}$ of functions $f$ with partial derivatives $\partial_\alpha f$ in $L^2$ for any tuple $\alpha = (\alpha_1, \ldots, \alpha_d)$ with $\max_\nu \alpha_\nu \leq s$.

We consider spaces $MB^s_q(L^r)$ such that $s > (1/\tau - 1/2)_+$, which are embedded in $L^2$ and strictly above the critical embedding line (with $\tau < 2$ and $s < 1/\tau - 1/2$, spaces $MB^s_q(L^r)$ are not embedded in $L^2$).
Theorem 6.6 (Minimax rates for Besov spaces $MB_q^s(L^r)$ with mixed dominating smoothness). Assume the target function $f^* \in MB_q^s(L^r)$ with $s > (1/\tau - 1/2)_+ \text{ and } 0 < q \leq \tau$. Then for sufficiently large $n$, the estimators using full or sparse tensor networks respectively satisfy
\[
\mathbb{E}\|\hat{f}_m^S - f^*\|^2_{2,\mu} \lesssim n^{-\frac{2\tilde{s}}{\tau}} \log(n)^{\frac{4}{\tau^2}}
\]
and
\[
\mathbb{E}\|\hat{f}_m^F - f^*\|^2_{2,\mu} \lesssim n^{-\frac{2\tilde{s}}{\tau}} \log(n)^{\frac{4}{\tau^2}}
\]
with $\tilde{s} = s$ if $k \geq s - 1/2$ or an arbitrary $\tilde{s} < s$ if $k < s - 1/2$.

Proof. See supplementary materials.

For $s > (1/p - 1/2)_+$, it is known that the minimax rate is lower bounded by $n^{-\frac{2\tilde{s}}{\tau}}$ (up to a logarithmic term) [40]. Therefore, Theorem 6.6 implies that our model selection strategy using sparse tensor networks achieves a rate arbitrarily close to minimax, up to a logarithmic term. With full tensor networks, the rate is close to minimax but slightly worse. We emphasize that this result is valid for any value of $k$, including $k = 0$. However, by adapting the degree $k$ to the regularity (i.e., $k \geq s - 1/2$), sparse tensor networks even achieve exactly the minimax rate.

Note that for $p \geq 2$, linear estimators based on hyperbolic cross approximation [17] achieve minimax rates, with a suitable choice of univariate approximation tools adapted to the regularity. Let us finally mention that for $p \geq 2$ and full tensor networks, by using [3, Theorem 6.6], we can obtain a slightly better rate in $n^{-\frac{2\tilde{s}}{\tau}}$ with $1 < C(d) < 2$.

6.3.3. Anisotropic Besov spaces

We now consider anisotropic Besov spaces $AB_q^\alpha(L^r)$, $\alpha = (s_1, \ldots, s_d) \in \mathbb{R}_+^d$, where $s_\nu > 0$ is related to the regularity order with respect to the $\nu$-th coordinate (see [3, 33] for a definition based on directional moduli of smoothness and the characterization of these spaces). For $\alpha = (s, \ldots, s)$ with $s > 0$, $AB_q^\alpha(L^r)$ coincides with the isotropic Besov space $B_q^s(L^r)$. For a tuple $\alpha$, we let $s(\alpha) := d(s_1^{-1} + \ldots + s_d^{-1})^{-1}$ be the aggregated smoothness parameter, such that $s_\nu := \tilde{s} \leq s(\alpha) \leq \bar{s} := \max_\nu s_\nu$.

We consider spaces $AB_q^\alpha(L^r)$ with $\alpha$ such that $s(\alpha) > d(1/\tau - 1/2)_+$, which are embedded in $L^2$. For these spaces, the minimax rate is in $n^{-\frac{2\tilde{s}}{\tau(s(\alpha))}}$ [35] and this rate can be achieved by linear estimators only for $\tau \geq 2$.

Theorem 6.7 (Minimax rates for anisotropic Besov spaces $AB_q^\alpha(L^r)$). Assume the target function $f^* \in AB_q^\alpha(L^r)$ with $\alpha \in \mathbb{R}_+^d$, such that $s(\alpha) > d(1/\tau - 1/2)_+$ and $0 < q \leq \tau$. Then for sufficiently large $n$, the estimators using full or sparse tensor networks respectively satisfy
\[
\mathbb{E}\|\hat{f}_m^S - f^*\|^2_{2,\mu} \lesssim n^{-\frac{2\tilde{s}}{\tau}} \log(n)^{\frac{4}{\tau^2}}
\]
and
\[
\mathbb{E}\|\hat{f}_m^F - f^*\|^2_{2,\mu} \lesssim n^{-\frac{2\tilde{s}}{\tau}} \log(n)^{\frac{4}{\tau^2}}
\]
with $\tilde{s} = s(\alpha)$ if $k \geq s - 1/2$ or an arbitrary $\tilde{s} < s$ if $k < s - 1/2$. 

Proof. See supplementary materials.

Theorem 6.7 implies that our model selection strategy using sparse tensor networks achieves a rate arbitrarily close to minimax, up to a logarithmic term. With full tensor networks, the rate is close to minimax but slightly worse. We again emphasize that this result is valid for any $k \in \mathbb{N}$, including $k = 0$. However, by adapting the degree $k$ to the highest regularity $\bar{s}$ (i.e., $k \geq 1/2$), sparse tensor networks even achieve exactly the minimax rate (up to the logarithmic term).

Let us mention that for $p \geq 2$ and full tensor networks, by using [3, Theorem 6.6], we can obtain a slightly better rate in $n^{-2\bar{s}/(2\bar{s}+C(d)d)}$ with $1 < C(d) < 2$.

Note that with a sufficient anisotropy such that $\sum_{d=1}^{d} s\nu^{-1} \leq \beta^{-1}$ with $\beta$ independent of $d$, we have $s(\alpha) \geq d\beta^{-1}$, and for an arbitrary $\tilde{\beta} < \beta$, our strategy with sparse (resp. full) tensor networks achieves a rate in $n^{-2\tilde{\beta}/(2\tilde{\beta}+1)}$ (resp. $n^{-\tilde{\beta}/(\tilde{\beta}+1)}$), which is independent of the dimension $d$.

6.3.4. Analytic functions.

Here, we consider the case of analytic functions on a bounded interval. We restrict the analysis to functions defined on $[0, 1]$ but the result could be easily extended to the multivariate case.

**Theorem 6.8 (Analytic functions).** Assume $f^* : [0, 1] \to \mathbb{R}$ admits an analytic extension on an open complex domain including $[0, 1]$. Then for sufficiently large $n$,

$$E\|\hat{f}_m^F - f^*\|_{2, \mu}^2 \lesssim n^{-1} \log(n)^{5/2}$$

up to logarithmic terms.

**Proof.** See supplementary materials.

The rate in $n^{-1}$ (up to logarithmic terms) achieved by full tensor networks is known to be the minimax rate for analytic functions for nonparametric estimation of analytic densities [8].

6.4. Beyond smoothness classes

We have seen that the proposed strategy is (near to) minimax adaptive to a large range of classical smoothness classes. In [3, Theorem 6.9], it is proved that for any $\alpha > 0$ and any $s > 0$, it holds

$$\mathcal{F}_q^s(L^2) \not\subset B_q^s(L^2),$$

that means that functions in the approximation classes of tensor networks do not need to have any smoothness in a classical sense. Tensor networks may thus achieve a good performance for functions that can not be captured by standard approximation tools.
such as splines or wavelets. That reveals the potential of tensor networks to achieve approximation or learning tasks for functions beyond standard smoothness classes. In particular, they have the potential to achieve a good performance in high-dimensional approximation tasks for function classes not described in terms of standard weighted or anisotropic smoothness.

Note that in [2, Proposition 5.21], it is proved that when limiting the resolution $L$ to be logarithmic in the complexity $c$ (i.e. when considering for $\Phi_c$ models for which $L = O(\log(c))$), the resulting approximation classes of tensor networks are continuously embedded in some Besov spaces. This highlights the importance of the resolution (or depth of the tensor network). Addressing learning tasks for functions beyond regularity classes requires to explore model classes with higher resolutions (i.e. with resolutions $L$ higher than $O(\log(c))$ and up to $c$).

Let us finally recall that the results of Section 6.3 have been obtained for tensor networks with variable resolution but a fixed tree at each resolution (corresponding to the tensor train format). Adaptiveness to a wide range of smoothness classes is thus achieved without tree adaptation. Much larger approximation classes are obtained by considering tensor networks with variables trees. In many high-dimensional applications, adapting the tree to the target function is necessary to achieve a good performance and circumvent the curse of dimensionality. Working with variable trees may thus be relevant to approximate highly structured functions beyond classical anisotropic smoothness spaces. Of course, this comes at a price of a higher computational complexity and requires in practice some exploration strategies as discussed in Section 7.

7. Discussion on practical aspects

The practical implementation of the proposed methodology requires a calibration of the penalty function. For that, we can rely on the slope heuristics method proposed by Birgé and Massart [9]. See [7] and [4] for a general presentation of the method. This method has shown very good performances and comes with mathematical guarantees in various settings. For non parametric Gaussian regression with i.i.d. error terms, see [9, 4] and references therein. The slope heuristics have several versions (see [4]), a version suitable for our setting is presented in supplementary materials.

Also, the exploration of all possible model classes $M^T_r(V)$ (or $M^{T,R}_r(V)$) with a complexity bounded by some $c$ is intractable since the number of such models is exponential in the number of variables $d$. Therefore, strategies should be introduced to propose a set of candidate model classes $M_m$, $m \in M$. In practice, a possible approach is to rely on adaptive learning algorithms from [21] (see also [20]) that generate predictors $f_m$ (minimizing the empirical risk) in a sequence of model classes $M^T_r(V)$. Note that developing an exploration strategy for sparse tensor networks is more challenging. These exploration strategies and practical learning algorithms are presented in supplementary materials.

The above strategies are heuristics approaches that have shown a good performance on some numerical experiments, as illustrated in supplementary materials.
References

Oct 2018.


