Statistical Learning in Tree-Based Tensor Format

Erwan Grelier[†]

Centrale Nantes
GeM (UMR CNRS 6183)

Mathilde Chevreuil

Université de Nantes
GeM (UMR CNRS 6183)

Anthony Nouy

Centrale Nantes
LMJL (UMR CNRS 6629)

[†]erwan.grelier@ec-nantes.fr

Statistical learning for uncertainty quantification

We denote by

- $X = (X_1, \dots, X_d)$ a set of d random variables,
- $Y = f(X)$ the random output of a model f , costly to evaluate.

The uncertainty quantification requires many realizations of Y , hence many evaluations of f . To reduce their cost we use an approximation v of f , by solving

$$\min_{v \in \mathcal{V}} \frac{1}{N} \sum_{k=1}^N |y^k - v(x^k)|^2 \quad (1)$$

with $\mathcal{V}$ an approximation set and (x^k, y^k) a realization of (X, Y) .

Tree-based (TB) tensor format

A function $v \in \mathcal{T}_r^T$ in tree-based tensor format associated to a dimension partition tree $T \subset 2^{\{1, \dots, d\}}$

admits a representation as a composition of multilinear vector-valued functions (tensors):

$$v(x) = v(x_1, x_2, x_3, x_4) = f_{1,2,3,4} \left(\underbrace{g_{1,2}(x_1, x_2)}_{f_{1,2}(f_1(x_1), f_2(x_2))}, \underbrace{g_{3,4}(x_3, x_4)}_{f_{3,4}(f_3(x_3), f_4(x_4))} \right)$$

where for the leaves of T , $1 \leq \nu \leq d$,

$$f_\nu : \mathcal{X}^\nu \rightarrow \mathbb{R}^{r_\nu},$$

and for a node α with s children $S(\alpha)$,

$$f_\alpha : \bigotimes_{\beta \in S(\alpha)} \mathbb{R}^{r_\beta} \rightarrow \mathbb{R}^{r_\alpha}$$

is a multilinear function identified with a tensor in $\mathbb{R}^{r_\alpha \times r_{\beta_1} \times \dots \times r_{\beta_s}}$.

Properties of the set $\mathcal{T}_r^T$:

- closed, best approximation problems are well posed,
- storage complexity of v scaling linearly with d ,
- multilinear parametrization of v ,
- existence of a higher-order singular value decomposition (HOSVD) of v .

Statistical learning in TB tensor format

Using the multilinear parametrization of $v \in \mathcal{T}_r^T$, problem (1) is solved using an alternating minimization algorithm, leading to linear problems:

$$\min_{a_\alpha \in \mathbb{R}^{m_\alpha}} \frac{1}{N} \sum_{k=1}^N |y^k - \Psi_\alpha(x^k)^\top a_\alpha|^2, \quad \forall \alpha \in T$$

with $\Psi_\alpha(x)$ such that $v(x) = \Psi_\alpha(x)^\top a_\alpha$. For the leaves of T , a sparse approximation is computed by using a working-set strategy.

Iterative adaptation of r

Starting from $r^0 = (0, \dots, 0)$, at iteration n , given an approximation $v^n \in \mathcal{T}_{r^n}^T$:

- increase the ranks $(r_\beta^n)_{\beta \in T_n}$ associated with a subset of nodes $T_n \subset T$,
- T_n is chosen as

$$T_n = \left\{ \alpha \in T : \sigma_\alpha \geq \theta \max_{\beta \in T} \sigma_\beta \right\},$$

with $\theta \in [0, 1]$ and σ_α the smallest α -singular value for the node α , obtained by computing the SVD of an α -matricization of v^n .

First numerical experiment

Approximation of

$$f(X) = \left(1 + \sum_{i=1}^d a_i X_i \right)^{-2},$$

- $X = (X_1, \dots, X_{10})$, $X_i \sim \mathcal{U}(0, 1)$,
- $0 \leq a_i \leq 1$, $i = 1, \dots, d$,
- linear tree (tensor train Tucker format),
- approximation spaces at the leaves: polynomial spaces of maximal degree 20,
- 10000 training samples.

TB learning and changes of variables

An approximation is searched in the form

$$v(x) = h(g(x)) = h(g_1(x), \dots, g_m(x)),$$

with $h \in \mathcal{T}_r^T$ and $g_i : \mathbb{R}^d \rightarrow \mathbb{R}$, $i = 1, \dots, m$.

Outline of the algorithm

Construction of approximations v_m , at each iteration introducing a new variable $z_m = g_m(x)$, and alternatively optimizing on

- $h \in \mathcal{T}_{r_m}^T$, with fixed g , using the statistical learning algorithm in TB tensor format,
- g , with fixed h , using a Gauss-Newton algorithm,

until convergence.

Second numerical experiment

Approximation of

$$f(X) = \sin(w_2^\top X + w_3^\top X) \cos(w_1^\top X + w_4^\top X) + \cos(w_1^\top X + w_3^\top X),$$

- $d = 20$,
- $X = (X_1, \dots, X_d)$,
- $X_i \sim \mathcal{U}(-1, 1)$, $i = 1, \dots, d$,
- $m \leq 4$,
- approximation spaces at the leaves: polynomial spaces of maximal degree 10,
- approximation space for the g_j , $j = 1, \dots, 4$: linear functions,
- $N = 1000$ training samples.

v	Confidence interval for the error	Storage complexity
h	$[0.00002, 1.1] \cdot 10^{-1}$	435
$h \circ g$	$[1.6, 7.9] \cdot 10^{-3}$	342

N being small compared to d , introducing changes of variables reduces the mean and variance of the error, and the storage complexity.

Conclusions

The statistical learning in tree-based tensor format:

- uses the classical machinery of linear approximation,
- exploits both low-rank and sparsity,
- can be combined with changes of variables.

Outlooks

- Perform tree adaptation,
- use other trees for $h \in \mathcal{T}_r^T$ when $v = h \circ g$,
- characterize the properties of the set of functions of the form $h \circ g$, $h \in \mathcal{T}_r^T$.

References

- [1] M. Chevreuil, R. Lebrun, A. Nouy, and P. Rai. A least-squares method for sparse low rank approximation of multivariate functions. *SIAM/ASA JUQ*, 2015.
- [2] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM Journal on Matrix Analysis and Applications*, 31(4):2029–2054, 2010.
- [3] W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*. Springer Berlin Heidelberg, 2012.
- [4] A. Nouy. Low-rank methods for high-dimensional approximation and model order reduction. In *Model Reduction and Approximation: Theory and Algorithms*. SIAM, Philadelphia, 2016.