Boosted least-squares and principal component analysis for training tree tensor networks

Cécile Haberstich[1,2], **Anthony Nouy**[2], **Guillaume Perrin**[1]

Mascot-Num 2020 | 17-18 September 2020

[1] CEA,DAM,DIF, F-91297, Arpajon, France,
[2] Centrale Nantes, LMJL UMR CNRS 6629, France

Approximation of high-dimensional functions

Context : Uncertainty quantification for a black-box and costly model represented by a function $u(x)$ of d variables.

Objective : Construct an approximation $u^\star$ of u in some model class V

→ with **controlled precision** (when $u \in L^2_\mu$, $\|u - u^\star\|_{L^2_\mu} \leq \varepsilon$),

→ with **only few evaluations** of $u(x^i)$ of u at points x^i chosen adaptively.

Difficulties : For a high dimension d,

→ V is an approximation space that should be adapted to the function u.
A typical choice is a tensor product space $V = V_1 \otimes \cdots \otimes V_d$, where each V_i is a suitable space of univariate functions.

→ When $d >> 1$ (even when each V_i is low-dimensional) → **curse of dimensionality**.

Approach

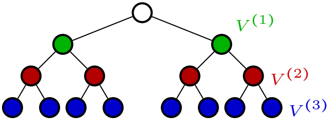
→ Here we propose a strategy to construct a nested sequence of well-chosen tensor product subspaces with decreasing dimensions, associated to a dimension partition tree T,

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

such that the approximation is defined by $u^\star = P_{V^\star} u$.

→ The resulting approximation is in **tree-based tensor format.**
It admits a multilinear parametrization with parameters forming a tree network of low-order → also called tree tensor networks.

→ The $V^{(i)}$ are constructed from the leaves of the tree to the root thanks to an extension of **Principal Component Analysis** to multivariate functions and **sample-based projections**.



Outline

Least-squares methods

In this part, we consider a linear space $V \subset L^2_\mu$ and $\{\varphi_j\}_{j=1}^m$ a given orthonormal basis of V. The best approximation of u by an element of V is given by the orthogonal projection :

$$P_V u = \arg\min_{v \in V} \|u - v\|^2_{L^2_\mu}.$$

- Since it is not computable in practice, replaced by a weighted least-squares projection :

$$\hat{P}_V u = \arg\min_{v \in V} \frac{1}{n} \sum_{i=1}^{n} w(x^i)(v(x^i) - u(x^i))^2 \text{ where } x^i \sim \rho$$

- The stability of the projection $\hat{P}_V$ is measured by the properties of the empirical Gram matrix $\hat{\boldsymbol{G}}$, more precisely by $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$.
- How to choose ρ to have the $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$ close to 0 while using a small n ?

Optimal least-squares methods

Theorem (Optimal weighted least-squares)

Let $d\rho = w(x)^{-1} d\mu(x)$ *with* $w(x)^{-1} = \frac{1}{m}\sum_{j=1}^{m} \varphi_j(x)^2$.
Let $\eta \in (0,1)$ *and* $\delta \in (0,1)$, *and for* $x^1, \cdots, x^n$ *i.i.d from* $d\rho$. *For* $n \geq \delta^{-2} m \log(2m\eta^{-1})$, *it holds*

$$\mathbb{P}(\|\hat{\boldsymbol{G}} - \boldsymbol{I}\| > \delta) \leq \eta.$$

The approximation $\hat{P}_V^C u$ *defined by* $\hat{P}_V u$ *if* $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\| < \delta$ *and* 0 *otherwise satisfies*

$$\mathbb{E}(\|u - \hat{P}_V u\|^2) \leq (1-\delta)^{-1} \|u - P_V u\|^2 + \eta \|u\|^2.$$

- ☹ Improving stability (smaller δ) and the chance to have this stability (smaller η) implies higher n.
- ☹ n still high compared to an interpolation method ($n = m$).

- Next, we propose a new measure $\tilde{\rho}$ based on ρ to improve the properties of $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$.

[2] A. Cohen and G. Migliorati. Optimal weighted least-squares methods. SMAI Journal of Computational Mathematics. 2017

Boosted optimal least-squares method (BLS)

1. **Resampling :** draw M independent n-samples $\{\boldsymbol{x}^{n,i}\}_{i=1}^{M}$, with $\boldsymbol{x}^{n,i} = (x^{1,i}, \cdots, x^{n,i})$, for each $1 \le j \le n$, $x^{j,i} \sim \rho$ and choose the one which minimizes $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$.

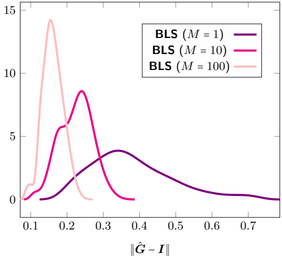


FIGURE – Distribution of $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$ for $\delta = 0.9$

Resampling improves the chance to be stable for a given δ ($\eta \to \eta^M$).

Boosted optimal least-squares method (BLS)

1. **Resampling :** draw M independent n-samples $\{\boldsymbol{x}^{n,i}\}_{i=1}^{M}$, with $\boldsymbol{x}^{n,i} = (x^{1,i}, \cdots, x^{n,i})$, for each $1 \leq j \leq n$, $x^{j,i} \sim \rho$ and choose the one which minimizes $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$.

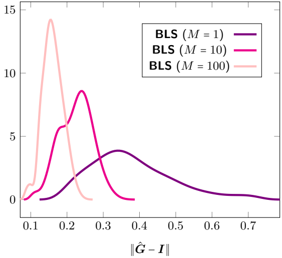


FIGURE – Distribution of $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$ for $\delta = 0.9$

Resampling improves the chance to be stable for a given δ ($\eta \to \eta^M$).

2. **Conditioning by rejection :** Repeat step 1 while $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\| > \delta$.

Boosted optimal least-squares method (BLS)

3. **Greedy removal of samples :** Begin with $K = \{1, \cdots, n\}$ and while $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\| \leq \delta$ successively select a subsample of size $\#K - 1$ which minimizes $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$.

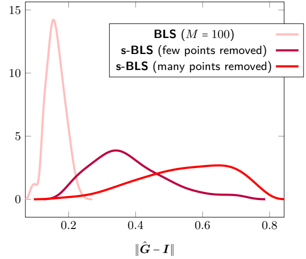


FIGURE – Distribution of $\|\hat{\boldsymbol{G}} - \boldsymbol{I}\|$ for $\delta = 0.9$

Boosted optimal least-squares (BLS)

Theorem (Stability of the boosted optimal least-squares)

Let $\eta \in (0,1)$ and $\delta \in (0,1)$, and let $\hat{P}_V u$ be the boosted optimal least-squares projection such that the initial sample size verifies $n \geq \delta^{-2} m \log(2m\eta^{-1})$ and the resulting number of samples after the greedy subsampling is constrained to be greater than n_0. It satisfies the quasi-optimality property

$$\mathbb{E}(\|u - \hat{P}_V u\|^2) \leq C \|u - P_V u\|^2$$

with $C = (1 + \frac{n}{n_0}(1-\delta)^{-1}(1-\eta^M)^{-1} M)$.

Also, assuming $\|u\|_{\infty,w} \leq L$, we can obtain a better bound.

For more details → see [3] C. Haberstich, A. Nouy, G. Perrin. Boosted optimal least-squares method. arXiv :1912.07075.

- ☺ quasi-optimality property
- ☹ pay the M and $\frac{n}{n_0}$

Illustration on a simple example : stability guaranteed

$$u(x) = \frac{1}{1 - \frac{0.5}{2d}\sum_{i=1}^{d} x_i}$$ defined on $\mathcal{X} = [-1,1]^d$, equipped with the uniform measure

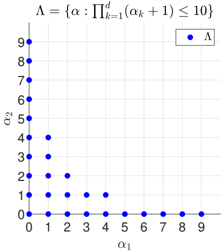


FIGURE – V is defined by a hyperbolic cross $d = 2$.

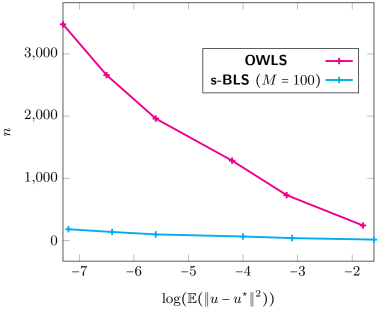


FIGURE – Guaranteed stability with probability greater than 0.99, $\delta = 0.9$.

Illustration on a simple example : given cost

$$u(x) = \frac{1}{1 - \frac{0.5}{2d}\sum_{i=1}^{d} x_i}$$ defined on $\mathcal{X} = [-1,1]^d$, equipped with the uniform measure

We have access to $u(x) + e$ with $e \sim \mathcal{N}(0, \sigma^2)$

- Given cost $n = m$
- Interpolation : $u^\star(x^i) = u(x^i) + e^i$ for $1 \le i \le m, x^i \in \mathcal{X}$, for example magic points. → interpolation may not be stable !

		Interpolation with magic points	s-BLS $(M = 100)$
m	σ	$\log(\|u - u^\star\|^2)$	$\|u - u^\star\|^2$
10	0.1	[-1.1 ; -1.0]	[-1.6 ; -1.1]
27	0.1	[-0.8 ; 0.1]	[-1.8 ; -0.7]
27	0.01	[-2.5 ; -1.5]	[-3.0 ; -2.3]

TABLE – Confidence intervals of levels 10% and 90% for the approximation error $\log(\|u - u^\star\|^2)$ for a noisy example with $d = 2$, $n = m$

Conclusions of the first part

The **boosted least-squares projection** is

- ☺ **stable in expectation**,
- ☺ with a number of samples **close to the dimension of the space** (almost the cost of an interpolation method),
- ☹ **error bound pessimistic** compared to the experiments.
- ☹ Sampling from the boosted optimal measure is **time-consuming**.
  (Remedies are sequential sampling for multivariate distributions, introduce an approximate greedy algo based on results in linear algebra).
- → However, when one evaluation of u is costly, this method is relevant.

Outline

Leaves-to-root strategy

- We consider the following dimension tree T,

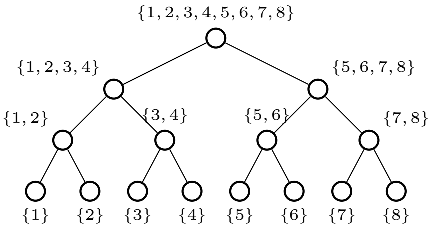


FIGURE – Dimension partition tree
$T = \{\{1,2,3,4,5,6,7,8\},\{1,2,3,4\},\{5,6,7,8\},\{1,2\},\{3,4\},...,\{7,8\},\{1\},\cdots,\{8\}\}$

- One node α is associated to a space of functions of groups of variables $x_\alpha = (x_i)_{i\in\alpha}$.

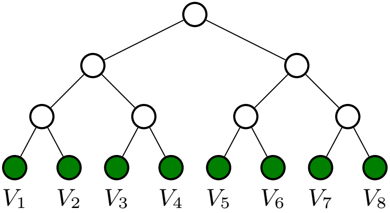
Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

and compute the approximation by porjecting u in $V^\star$.

- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.



$$V^{(4)} = \bigotimes_{i=1}^{8} V_i$$

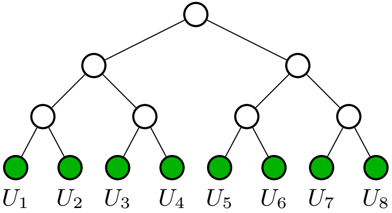
Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

and compute the approximation by porjecting u in $V^\star$.

- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.



For each α, $U_\alpha \subset V_\alpha$ $V^{(3)} = \bigotimes_{i=1}^8 U_i$

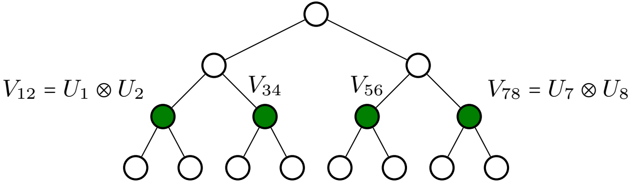
Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

and compute the approximation by porjecting u in $V^\star$.

- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.

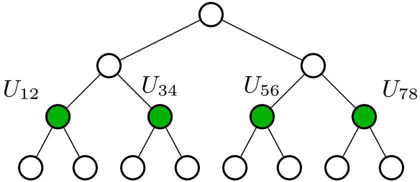


Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

  and compute the approximation by porjecting u in $V^\star$.
- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.



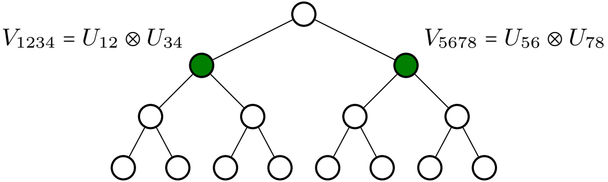
For each α, $U_\alpha \subset V_\alpha$ $V^{(2)} = U_{12} \otimes U_{34} \otimes U_{56} \otimes U_{78}$

Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

  and compute the approximation by porjecting u in $V^\star$.
- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.



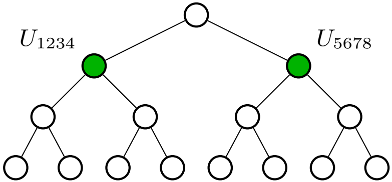
Leaves-to-root strategy

- Introduce a finite-dimensional approximation space $V = V_1 \otimes V_2 \otimes \cdots \otimes V_8 \subset L^2_\mu$.
- Construct a nested sequence of well-chosen subspaces

$$V = V^{(L)} \supset \cdots \supset V^{(2)} \supset V^{(1)} = V^\star,$$

and compute the approximation by porjecting u in $V^\star$.

- More precisely, going from the leaves to the root, construct a hierarchy of low-dimensional subspaces $(U_\alpha)_{\alpha \in T}$ associated to the tree T which defines the sequence $V^{(i)}$.



For each α, $U_\alpha \subset V_\alpha$ and $V^{(1)} = V^\star = U_{1234} \otimes U_{5678}$

- Final approximation is given by $u^\star = \hat{P}_{V^\star} u$ with $V^\star = U_{1234} \otimes U_{5678}$.

How to construct near-optimal subspaces U_α ?

- A multivariate function can be identified with a bivariate function.
- The truncated singular value decomposition u_{r_α} of u :

$$u_{r_\alpha}(x_\alpha, x_{\alpha^c}) = \sum_{i=1}^{r_\alpha} \sigma_i v_i^\alpha(x_\alpha) v_i^{\alpha^c}(x_{\alpha^c})$$

is the solution of the problem of best approximation of u by a function with α-rank r_α

$$\min_{\text{rank}_\alpha(v) \le r_\alpha} \|u - v\|^2$$

- $v_1^\alpha, \cdots, v_{r_\alpha}^\alpha$ are the r_α α-principal components of u and $U_\alpha = \text{span}\{v_1^\alpha, \cdots, v_{r_\alpha}^\alpha\}$ is the α-principal subspace of u.

  In practice to estimate U_α two approximations are made :

1. Statistical estimation of the α-principal subspaces with an **adaptive strategy based on cross validation**.
2. Compute the α-principal subspace of a **projection of** u. (using BLS).

Error bound

- The final approximation $u^\star$ verifies :

$$\mathbb{E}(\|u-u^\star\|^2) \leq \sum_{\alpha \in T \setminus \text{root}} (2C)^{l(\alpha)} \varepsilon_{pca}^2(\alpha) + \sum_{\alpha \in \text{leaves}} \frac{1}{2}(2C)^{l(\alpha)+1} \varepsilon_{dis}^2(\alpha)$$

- C is the quasi-optimality constant from the boosted least-squares projection.

In theory, if we want a **controlled approximation** $\mathbb{E}(\|u-u^\star\|^2) \leq \varepsilon^2$, we have to

→ **Adapt ranks** and **control the estimation** of U_α such that

$$\varepsilon_{pca}^2(\alpha) \leq \frac{\varepsilon^2}{(2C)^{l(\alpha)}(\#T-1)}$$

→ and also, **control the discretization error**,

$$\varepsilon_{dis}^2(\alpha) \leq \frac{\varepsilon^2}{\frac{1}{2}(2C)^{l(\alpha)+1} d}.$$

→ But, C is large and $l(\alpha)$ may be high (for high d and deep trees), in practice we assume this bound is not sharp and use heuristics to control the error (cross validation).

Illustration of the choice of the projection

- Let $u(x) = \sin(x_1 + \cdots + x_{10})$ and $\mathcal{X} = \mathbb{R}^{10}$ equipped with the gaussian measure.
- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen such that there is a negligeable discretization error ($p = 20$).
- T is a balanced binary tree.
- Approximation with prescribed tolerance $\varepsilon = 10^{-9}$.

Interpolation		Boosted least-squares	
$\log(\sqrt{\mathbb{E}(\|u - u^\star\|^2)})$	n	$\log(\sqrt{\mathbb{E}(\|u - u^\star\|^2)})$	n
-8.5	$[1110; 4405]$	-9.2	$[940; 946]$

TABLE – $\log(\sqrt{\mathbb{E}(\|u - u^\star\|^2)})$ and confidence intervals of levels 10% and 90% for the number of evaluations n.

Illustration of the adaptive strategy for the estimation of the α-principal components

- Let $u(x) = \frac{1}{(10+2x_1+x_3+2x_4-x_5)^2}$ and $\mathcal{X} = [-1,1]^d$ equipped with the uniform measure.
- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen adaptively to reach a negligeable discretization error ($p \leq 15$) using adaptive boosted least-squares.
- T is a balanced binary tree.

With adaptive strategy for PCA

ε	$\log(\sqrt{\mathbb{E}(\|u-u^\star\|^2)})$	n
-2	-3	[328 ; 403]
-3	-4.1	[455 ; 579]
-4	-4.4	[534 ; 697]
-5	-5.3	[751 ; 985]
-6	-6.1	[1028 ; 1503]
-7	-7.0	[1463 ; 2230]

TABLE – Heuristic control of the precision. $\log(\sqrt{\mathbb{E}(\|u-u^\star\|^2)})$ (in log scale) and confidence intervals of levels 10% and 90% for the number of evaluations n.

Conclusions of the second part

The **tree-based tensor format approximation**

- ☺ with **BLS** guarantees **stability** for the final approximation (compared to interpolation),
- ☺ estimation of the α-principal components can be controlled through **adaptive strategies** (with a near-optimal number of evaluations, only observed, no theory)
- ☺ **final approximation with a controlled error** (if we pay the price ...).
- ☹ Computing **BLS** projectors requires many samples from and multivariate measures (same remedies as before).
- ☹ The α-ranks may be large for a given tree T.

Outline

A motivating example

- $\mathcal{X} = [-1,1]^d$, equipped with the uniform measure and the function u defined as follows,

$$u(x) = g(x_1,x_2) + g(x_3,x_4) + \ldots + g(x_{d-1},x_d), \text{ where } g(x_\nu,x_{\nu+1}) = \sum_{i=0}^{3} x_\nu^i x_{\nu+1}^i.$$

- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen to have a negligeable discretization error ($p = 4$).

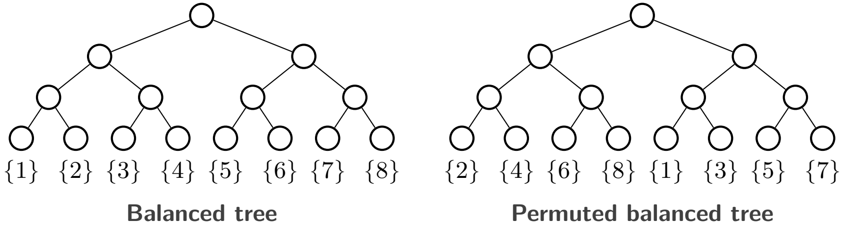


FIGURE – Two balanced trees, ordered variables (left) and permuted variables (right).

A motivating example

- $\mathcal{X} = [-1,1]^d$, equipped with the uniform measure and the function u defined as follows,

$$u(x) = g(x_1,x_2) + g(x_3,x_4) + \ldots + g(x_{d-1},x_d), \text{ where } g(x_\nu,x_{\nu+1}) = \sum_{i=0}^{3} x_\nu^i x_{\nu+1}^i.$$

- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen to have a negligeable discretization error ($p = 4$).

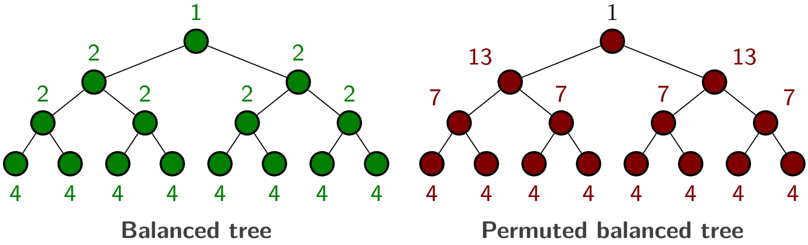


FIGURE – Two balanced trees, ordered variables (left) and permuted variables (right), with the α-ranks

A motivating example

- $\mathcal{X} = [-1,1]^d$, equipped with the uniform measure and the function u defined as follows,
$$u(x) = g(x_1,x_2) + g(x_3,x_4) + \ldots + g(x_{d-1},x_d), \text{ where } g(x_\nu,x_{\nu+1}) = \sum_{i=0}^{3} x_\nu^i x_{\nu+1}^i.$$
- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen to have a negligeable discretization error ($p = 4$).

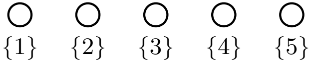
	Balanced tree	**Permuted balanced tree**
d	n	n
8	[460; 460]	[2293; 2438]
16	[940; 957]	[13679; 14682]
24	[1420; 1471]	[45921; 49402]

TABLE – Confidence intervals of levels 10% and 90% for the number of evaluations n with two different dimension partition trees.

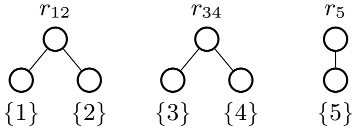
Leaves-to-root optimization of the tree

1. **For each leaf** $\alpha = \{\nu\}$**,** $1 \leq \nu \leq d$, we determine U_α an approximation of the α-principal subspace of u.

→ r_1, r_2, r_3, r_4 and r_5 are known.



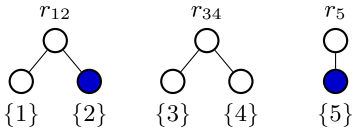
2. Choose a random pairing $\mathcal{P}$ and estimate the associated α-ranks



and calculate the corresponding cost function
$\mathcal{C} = \sum_{\alpha \in \mathcal{P}} r_\alpha r_{S_1(\alpha)} r_{S_2(\alpha)} = r_{12} r_1 r_2 + r_{34} r_3 r_4 + r_5^2$

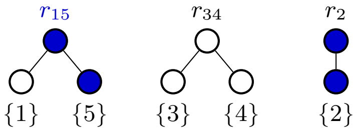
Leaves-to-root optimization of the tree

2. Select two nodes β_1 and β_2 (choosing preferentially the ones whose parent has a high α-rank), $\beta \sim \text{rank}_{parent(\beta)}(u)^\gamma$ with γ an integer.



Leaves-to-root optimization of the tree

2. Select two nodes β_1 and β_2 (choosing preferentially the ones whose parent has a high α-rank), $\beta \sim \text{rank}_{parent(\beta)}(u)^\gamma$ with γ an integer.

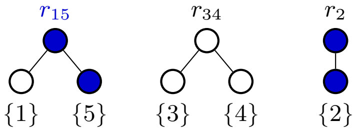


and permute these two nodes. Estimate the new α-ranks (associated to this new partition), calculate the new cost $\mathcal{C}^\star$, if $\mathcal{C}^\star < \mathcal{C}$ accept the permutation.

3. Repeat the operation n_P times.

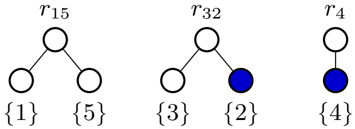
Leaves-to-root optimization of the tree

2. Select two nodes β_1 and β_2 (choosing preferentially the ones whose parent has a high α-rank), $\beta \sim \text{rank}_{parent(\beta)}(u)^\gamma$ with γ an integer.



and permute these two nodes. Estimate the new α-ranks (associated to this new partition), calculate the new cost $\mathcal{C}^\star$, if $\mathcal{C}^\star < \mathcal{C}$ accept the permutation.

3. Repeat the operation n_P times.

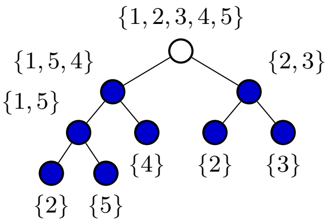


Determine U_α for $\alpha = \{1,5\},\{2,3\}$. $\rightarrow r_{15}, r_{23}$ and r_4 are known.

Leaves-to-root optimization of the tree

4. Proceed similarly with the next level, for pairing $\{1,5\}$, $\{2,3\}$ and $\{4\}$.

5. This yields a dimension tree.



6. Compute the final approximation $u^\star = \hat{P}_{U_{\{1,4,5\}} \otimes U_{\{2,3\}}} u$

Numerical example with local optimization

- $\mathcal{X} = [-1,1]^d$, with, $d = 24$, equipped with the uniform measure and the function u,

$$u(x) = g(x_1, x_2) + g(x_3, x_4) + \ldots + g(x_{d-1}, x_d), \text{ where } g(x_\nu, x_{\nu+1}) = \sum_{i=0}^{3} x_\nu^i x_{\nu+1}^i.$$

- Polynomial approximation spaces $V_\nu = \mathbb{P}_p(\mathcal{X}_\nu)$, with p chosen to have a negligeable discretization error ($p = 4$).
- Approximation with a prescribed tolerance $\varepsilon = 10^{-14}$

	n	n_{total}
	$[q_{10} ; q_{50} ; q_{90}]$	$[q_{10} ; q_{50} ; q_{90}]$
Deterministic algo from [1]	[1540 ; 2075 ; 3008]	[24221 ; 27182 ; 28313]
Stochastic algo presented here	[2955 ; 6321 ; 10814]	[9865 ; 14212 ; 19089]
Random Balanced Tree	[17867 ; 24115 ; 35865]	[17867 ; 24115 ; 35865]

TABLE – q_{10}, q_{50}, q_{90} are the $10^{th}, 50^{th}$ and 90^{th} quantiles for a number of evaluations n, $n_P = 10d$.

[1] Grasedyck L. Ballani J. Tree adaptive approximation in the hierarchical tensor format. SIAM J. Sci. Comput. 2014.

Conclusions of the third part

- ☹ Tree optimization is a **combinatorial problem**.
- ☺ **Stochastic algorithm** → compromise between the number of trees explored (cost for optimization) and the search of the optimum, compared to a deterministic strategy.
- ☺ Total cost is **better in expectation than a random tree**.

Outline

Conclusions

The proposed algorithm :

- provides an **approximation of u in tree-based tensor format** using evaluations of the function at a structured set of points,
- provides a **controlled approximation** (for a sufficiently a high number of evaluations of the function u).
- Under some assumptions on the function class and results on empirical PCA, a **bound of the number of evaluations** necessary to reach a certain precision can be obtained (very pessimistic compared to experiments...).

We proposed **fully adaptive strategies** for :

- the control of the discretization error,
- the tree selection,
- the control of the α-ranks,
- the estimation of the principal components.

J. Ballani and L. Grasedyck.
Tree adaptive approximation in the hierarchical tensor format.
SIAM J. Sci. Comput., 36(4) :A1415–A1431. (17 pages), 2014.

A. Cohen and G. Migliorati.
Optimal weighted least-squares methods.
SMAI Journal of Computational Mathematics, 3 :181–203, 2017.

C. Haberstich, A. Nouy, and G. Perrin.
Boosted optimal weighted least-squares methods.
arXiv :1912.07075, 2019.

A. Nouy.
Higher-order principal component analysis for the approximation of tensors in tree-based low rank formats.
Numer. Math., 141(3) :743–789, 2019.

Thank you for your attention.
Do you have any questions ?