CEMRACS,
July 19-23, 2021

Approximation and learning with tensor networks

Part III: Computational aspects

Anthony Nouy

Centrale Nantes, Laboratoire de Mathématiques Jean Leray

Computing with tensor networks

We here present some algorithms for the approximation of tensors (or functions) using tensor networks.

Different contexts depending on the available information on the tensor:

- all entries of the tensor,
- equations satisfied by the tensor,
- some entries, either arbitrary or structured,
- more general functionals of the tensor.

Available software

- tensap. A Python package for the approximation of functions and tensors. (link to GitHub page).
- ApproximationToolbox. An object-oriented MATLAB toolbox for the approximation of functions and tensors. (link to GitHub page).

Outline

1. Higher-order singular value decomposition and tensor truncation
2. Learning from structured evaluations
3. Direct optimization in subsets of tensor networks
4. Iterative methods with tensor truncation
5. Thresholding of singular values and relaxation methods

Outline

1. Higher-order singular value decomposition and tensor truncation
2. Learning from structured evaluations
3. Direct optimization in subsets of tensor networks
4. Iterative methods with tensor truncation
5. Thresholding of singular values and relaxation methods

Hilbertian setting

We consider a tensor u in a Hilbert tensor space $V^1 \otimes \ldots \otimes V^d$ and we assume that u is given as a full tensor or in a certain low-rank format.

We present truncation schemes for finding a low-rank approximation of u with reduced complexity, relying on the standard singular value decomposition of order-two tensors.

We denote by $\|\cdot\|$ the canonical norm on $V^1 \otimes \ldots \otimes V^d$.

For an algebraic tensor in $\mathbb{R}^{I_1} \otimes \ldots \otimes \mathbb{R}^{I_d}$, $\|\cdot\|$ is the Frobenius norm

$$\|u\|^2 = \sum_{i_1 \in I_1} \ldots \sum_{i_d \in I_d} u(i_1, \ldots, i_d)^2$$

Truncated singular value decomposition for order-two tensors

An order-two tensor u in $V^1 \otimes V^2$ admits a singular value decomposition

$$u = \sum_{k \geq 1} \sigma_k v_k^1 \otimes v_k^2,$$

where the singular values $\sigma(u) = \{\sigma_k\}_{k \geq 1}$ are sorted by decreasing order.

An element of best approximation of u in the set of tensors with rank bounded by r is provided by the truncated singular value decomposition

$$u_r = \sum_{k=1}^{r} \sigma_k v_k^1 \otimes v_k^2,$$

with an error

$$\|u - u_r\|^2 = \min_{\mathrm{rank}(v) \leq r} \|u - v\|^2 = \sum_{k \geq r+1} \sigma_k^2.$$

Truncated singular value decomposition for order-two tensors

An approximation u_r with relative precision ϵ, such that

$$\|u - u_r\| \leq \epsilon \|u\|,$$

can be obtained by choosing a rank r such that

$$\sum_{k \geq r+1} \sigma_k^2 \leq \epsilon^2 \sum_{k \geq 1} \sigma_k^2.$$

The complexity of computing the singular value decomposition of a tensor u is $O(n^3)$ if $\dim(V^1) = \dim(V^2) = n$. If u is given in low-rank format $u = \sum_{k=1}^{R} a_k \otimes b_k$, with a rank $R < n$, the complexity breaks down to $O(R^3 + 2Rn^2)$.

Higher-order singular value decomposition

For a non-empty subset α in $D = \{1, \ldots, d\}$, a tensor $u \in V^1 \otimes \ldots \otimes V^d$ can be identified with its matricisation

$$\mathcal{M}_\alpha(u) \in V^\alpha \otimes V^{\alpha^c},$$

an order-two tensor which admits a singular value decomposition

$$\mathcal{M}_\alpha(u) = \sum_{k \geq 1} \sigma_k^\alpha v_k^\alpha \otimes w_k^{\alpha^c} \equiv u.$$

$\sigma^\alpha(u) := \{\sigma_k^\alpha\}_{k \geq 1}$ are the α-singular values of u.

The α-rank of u is the number of non-zero α-singular values

$$\mathrm{rank}_\alpha(u) = \|\sigma^\alpha(u)\|_0.$$

Higher-order singular value decomposition

By sorting the α-singular values by decreasing order, an approximation u_r with α-rank r can be obtained by retaining the r largest α-singular values, i.e.

$$u_r \equiv \sum_{k=1}^{r} \sigma_k^\alpha v_k^\alpha \otimes w_k^{\alpha^c},$$

The vectors $\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ are the dominant α-singular vectors of u or α-principal components of u.

The space $U_{r_\alpha}^\alpha = span\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ is the dominant α-principal subpace of u.

Higher-order singular value decomposition

By sorting the α-singular values by decreasing order, an approximation u_r with α-rank r can be obtained by retaining the r largest α-singular values, i.e.

$$u_r \equiv \sum_{k=1}^{r} \sigma_k^\alpha v_k^\alpha \otimes w_k^{\alpha^c},$$

The vectors $\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ are the dominant α-singular vectors of u or α-principal components of u.

The space $U_{r_\alpha}^\alpha = span\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ is the dominant α-principal subpace of u.

Denote by $P_{U_{r_\alpha}^\alpha}$ the orthogonal projection from V^α to $U_{r_\alpha}^\alpha$ and by $\mathcal{P}_{U_{r_\alpha}^\alpha} = P_{U_{r_\alpha}^\alpha} \otimes id_{\alpha^c}$ the orthogonal projection defined on V such that for $v^\alpha \otimes w^{\alpha^c} \in V^\alpha \otimes V^{\alpha^c}$,

$$\mathcal{P}_{U_{r_\alpha}^\alpha}(v^\alpha \otimes w^{\alpha^c}) = (P_{U_{r_\alpha}^\alpha} v^\alpha) \otimes w^{\alpha^c}$$

Higher-order singular value decomposition

By sorting the α-singular values by decreasing order, an approximation u_r with α-rank r can be obtained by retaining the r largest α-singular values, i.e.

$$u_r \equiv \sum_{k=1}^{r} \sigma_k^\alpha v_k^\alpha \otimes w_k^{\alpha^c},$$

The vectors $\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ are the dominant α-singular vectors of u or α-principal components of u.

The space $U_{r_\alpha}^\alpha = span\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$ is the dominant α-principal subpace of u.

Denote by $P_{U_{r_\alpha}^\alpha}$ the orthogonal projection from V^α to $U_{r_\alpha}^\alpha$ and by $\mathcal{P}_{U_{r_\alpha}^\alpha} = P_{U_{r_\alpha}^\alpha} \otimes id_{\alpha^c}$ the orthogonal projection defined on V such that for $v^\alpha \otimes w^{\alpha^c} \in V^\alpha \otimes V^{\alpha^c}$,

$$\mathcal{P}_{U_{r_\alpha}^\alpha}(v^\alpha \otimes w^{\alpha^c}) = (P_{U_{r_\alpha}^\alpha} v^\alpha) \otimes w^{\alpha^c}$$

We have

$$u_r = \mathcal{P}_{U_{r_\alpha}^\alpha} u$$

and

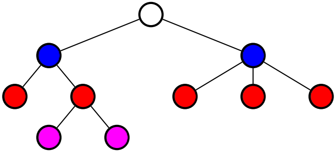
$$\|u - u_r\|^2 = \min_{\text{rank}_\alpha(v) \leq r} \|u - v\|^2 = \sum_{k>r} (\sigma_k^\alpha)^2.$$

Truncation scheme for tree-based tensor formats

For tree-based tensor formats

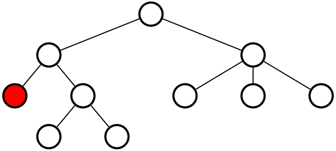
$$\mathcal{T}_r^T(V) = \{v \in V : \mathrm{rank}_\alpha(v) \leq r_\alpha, \alpha \in T\},$$

where T is a dimension partition tree over $D = \{1, \ldots, d\}$, different variants of higher order singular value decomposition (also called hierarchical singular value decomposition) can be defined from singular value decompositions of matricisations $\mathcal{M}_\alpha(u)$ of a tensor u.



Leaves to root truncation scheme for tree-based tensor formats

For each leaf node α, let $U^{\alpha}_{r_\alpha}$ be the r_α-dimensional α-principal subspace of u.

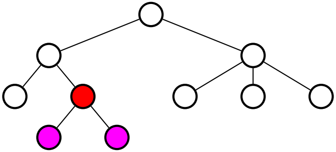


For each interior node $\alpha \in T \setminus \{D\}$ with children $S(\alpha)$, define a tensor space

$$V_\alpha = \bigotimes_{\beta \in S(\alpha)} U^{\beta}_{r_\beta}$$

and let $U^{\alpha}_{r_\alpha} \subset V_\alpha$ be the r_α-dimensional α-principal subspace of

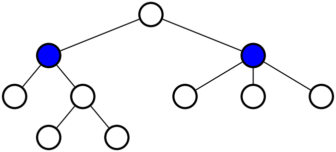
$$u_\alpha = \mathcal{P}_{V_\alpha} u$$



Leaves to root truncation scheme for tree-based tensor formats

Finally define u_r as the orthogonal projection onto the tensor space $V_D = \bigotimes_{\alpha \in S(D)} U_\alpha$

$$u_r = \mathcal{P}_r^{(1)} u = \mathcal{P}_r^{(1)} \dots \mathcal{P}_r^{(L)} u$$



Leaves to root truncation scheme for tree-based tensor formats

The obtained approximation u_r is such that

$$\|u - u_r\|^2 \leq \sum_{\alpha \in T \setminus D} \min_{\operatorname{rank}_\alpha(v) \leq r_\alpha} \|u - v\|^2 = \sum_{\alpha \in T \setminus D} \sum_{k_\alpha > r_\alpha} (\sigma^\alpha_{k_\alpha})^2,$$

from which we deduce that u_r is a quasi-optimal approximation of u in $\mathcal{T}_r^T$ such that

$$\|u - u_r\| \leq C(T) \min_{v \in \mathcal{T}_r^T} \|u - v\|,$$

where $C(T) = \sqrt{\#T - 1}$ is the square root of the number of projections applied to the tensor. The number of nodes of a dimension partition tree T being bounded by $2d - 1$,

$$C(T) \leq \sqrt{2d - 2}.$$

Leaves to root truncation scheme for tree-based tensor formats

The obtained approximation u_r is such that

$$\|u - u_r\|^2 \leq \sum_{\alpha \in T \setminus D} \min_{\mathrm{rank}_\alpha(v) \leq r_\alpha} \|u - v\|^2 = \sum_{\alpha \in T \setminus D} \sum_{k_\alpha > r_\alpha} (\sigma^\alpha_{k_\alpha})^2,$$

from which we deduce that u_r is a quasi-optimal approximation of u in $\mathcal{T}_r^T$ such that

$$\|u - u_r\| \leq C(T) \min_{v \in \mathcal{T}_r^T} \|u - v\|,$$

where $C(T) = \sqrt{\#T - 1}$ is the square root of the number of projections applied to the tensor. The number of nodes of a dimension partition tree T being bounded by $2d - 1$,

$$C(T) \leq \sqrt{2d - 2}.$$

Also, if we select the ranks $(r_\alpha)_{\alpha \in T \setminus D}$ such that for all α

$$\sum_{k_\alpha > r_\alpha} (\sigma^\alpha_{k_\alpha})^2 \leq \frac{\epsilon^2}{C(T)^2} \sum_{k_\alpha \geq 1} (\sigma^\alpha_{k_\alpha})^2 = \frac{\epsilon^2}{C(T)^2} \|u\|^2,$$

we finally obtain an approximation u_r with relative precision ϵ,

$$\|u - u_r\| \leq \epsilon \|u\|.$$

Leaves to root truncation scheme for tree-based tensor formats

If u is in some tensor space $W = W_1 \otimes \ldots \otimes W_d$ and $V = V_1 \otimes \ldots \otimes V_d$ is a finite-dimensional tensor subspace of W, an approximation in the tensor format $\mathcal{T}_r^T(V)$ can be obtained by modifying the procedure for the leaves.

For each leaf node α, $U^\alpha_{r_\alpha}$ is defined as a α-principal subspace of $u_\alpha = \mathcal{P}_{V_\alpha} u$.

Theorem (Fixed rank)

For a given T-rank, we obtain an approximation $u_r \in \mathcal{T}_r^T(V)$ such that

$$\|u_r - u\|^2 \leq C(T)^2 \min_{v \in \mathcal{T}_r^T} \|v - u\|^2 + \sum_{\text{leaves } \alpha} \|u - \mathcal{P}_{V_\alpha} u\|^2$$

Theorem (Fixed precision)

For a desired precision ϵ, if the α-ranks are determined such that

$$\|\mathcal{P}_{U^\alpha_{r_\alpha}} u_\alpha - u_\alpha\| \leq \frac{\epsilon}{C(T)} \|u_\alpha\|,$$

we obtain an approximation u_r such that

$$\|u_r - u\|^2 \leq \epsilon^2 \|u\|^2 + \sum_{\text{leaves } \alpha} \|u - \mathcal{P}_{V_\alpha} u\|^2.$$

Efficient truncation algorithms

Recent works for efficient truncation algorithms

- Randomized linear algebra [Che/Wei'19,Sun'20,Huber'17]
- Block-wise tensor compressions [Ehrlacher'21]
- Parallel algorithms [Grigori/Kumar'20,Daas'20]
- ...

Outline

1. Higher-order singular value decomposition and tensor truncation
2. Learning from structured evaluations
3. Direct optimization in subsets of tensor networks
4. Iterative methods with tensor truncation
5. Thresholding of singular values and relaxation methods

Learning from structured evaluations

For the approximation of a tensor (or function) in tree-based format from evaluations of the tensor at some entries, different strategies have been proposed, either based on cross approximation [Oseledets'10, Ballani'13] or principal component analysis [Nouy'19, Haberstich'21].

These methods rely on structured evaluations

$$u(x_\alpha^i, x_{\alpha^c}^j)$$

where x_α^i are samples of the variables x_α, and $x_{\alpha^c}^j$ samples of the variables x_{α^c}.

Learning from principal component analysis

Assume that $X = (X_1, \ldots, X_d)$ has a probability measure $\mu = \mu_1 \otimes \ldots \otimes \mu_d$ with support $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_d$.

Consider a multivariate function $u \in L^2_\mu(\mathcal{X})$ and assume that we can evaluate the function for arbitrary instance x of X.

For each a subset of variables α and its complementary subset $\alpha^c = D \setminus \alpha$, u is identified with a bivariate function which admits a singular value decomposition

$$u(x_\alpha, x_{\alpha^c}) = \sum_{k=1}^{\mathrm{rank}_\alpha(u)} \sigma_k^\alpha v_k^\alpha(x_\alpha) v_k^{\alpha^c}(x_{\alpha^c})$$

Learning from principal component analysis

The subspace of α-principal components

$$U_\alpha = span\{v_1^\alpha, \dots, v_{r_\alpha}^\alpha\}$$

is such that

$$u_{r_\alpha}(\cdot, x_{\alpha^c}) = \mathcal{P}_{U_\alpha} u(\cdot, x_{\alpha^c})$$

It is solution of

$$\min_{\dim(U_\alpha)=r_\alpha} \|u - \mathcal{P}_{U_\alpha} u\|^2$$

that is for $\|\cdot\|$ the $L^2_\mu(\mathcal{X})$-norm,

$$\min_{\dim(U_\alpha)=r_\alpha} \mathbb{E}\left(\|u(\cdot, X_{\alpha^c}) - \mathcal{P}_{U_\alpha} u(\cdot, X_{\alpha^c})\|^2_{L^2_{\mu_\alpha}(\mathcal{X}_\alpha)}\right)$$

where u is seen as a function-valued random variable

$$u(\cdot, X_{\alpha^c}) \in L^2_{\mu_\alpha}(\mathcal{X}_\alpha).$$

Learning algorithm based on principal component analysis

In order to construct an approximation in the tree-based format $\mathcal{T}_r^T(V)$, with V some feature tensor space, we apply the root to leaves procedure.

For a feasible algorithm using samples:

- Replacement of orthogonal projections by sampled-based projections.
- Statistical estimation of principal subspaces.

From orthogonal to sampled-based projections

Orthogonal projections $\mathcal{P}_{V_\alpha}$ on subspaces V_α are replaced by oblique projections $\mathcal{I}_{V_\alpha}$ using samples, typically interpolation or least-squares projection.

For a function u and a given value x_{α^c} of the group of variables X_{α^c},

$$\mathcal{I}_{V_\alpha} u(\cdot, x_{\alpha^c}) = \sum_{i=1}^{M_\alpha} a_i(x_{\alpha^c}) \psi_i^\alpha(\cdot)$$

where the ψ_i^α form a basis of V_α, and the coefficients $a_i(x_{\alpha^c})$ depend on evaluations $u(x_\alpha^k, x_{\alpha^c})$ for some samples x_α^k of X_α (interpolation points or random samples).

In practice,

- for interpolation, possible use of magic points x_α^i [Nouy '19],
- for least-squares projection, possible use of optimal weighted least-squares for a control of the norm of operators $\mathcal{I}_{V_\alpha}$ [Cohen/Migliorati'17,Habertisch '21].

Statistical estimation of principal subspaces

The α-principal subspaces U_α of $u_\alpha = \mathcal{I}_{V_\alpha} u$ are defined by

$$\min_{\dim(U_\alpha)=r_\alpha} \mathbb{E}\left(\|\mathcal{I}_{V_\alpha} u(\cdot, X_{\alpha^c}) - \mathcal{P}_{U_\alpha} \mathcal{I}_{V_\alpha} u(\cdot, X_{\alpha^c})\|^2_{L^2_{\mu_\alpha}(\mathcal{X}_\alpha)} \right)$$

Principal subspaces can be estimated using i.i.d. samples $u(\cdot, x^j_{\alpha^c})$ of this random variable and by solving

$$\min_{\dim(U_\alpha)=r_\alpha} \frac{1}{N_\alpha} \sum_{j=1}^{N_\alpha} \|\mathcal{I}_{V_\alpha} u(\cdot, x^j_{\alpha^c}) - \mathcal{P}_{U_\alpha} \mathcal{I}_{V_\alpha} u(\cdot, x^j_{\alpha^c})\|^2_{L^2_{\mu_\alpha}(\mathcal{X}_\alpha)}$$

where $\{x^j_{\alpha^c}\}_{j=1}^{N_\alpha}$ are i.i.d. samples of the group of variables X_{α^c}.

If the projection $\mathcal{I}_{V_\alpha}$ is based on a set of M_α samples of X_α, this requires the evaluation of u at the $M_\alpha \times N_\alpha$ points

$$\{(x^i_\alpha, x^j_{\alpha^c}) : 1 \le i \le M_\alpha, 1 \le j \le N_\alpha\}.$$

Outline

1. Higher-order singular value decomposition and tensor truncation

2. Learning from structured evaluations

3. Direct optimization in subsets of tensor networks

4. Iterative methods with tensor truncation

5. Thresholding of singular values and relaxation methods

Direct optimization in subsets of tensor networks

Consider a subset of tensors $\mathcal{M}_r$ that admits a multilinear parametrization of the form

$$v(x_1, \dots, x_d) = \sum_{k_1=1}^{r_1} \dots \sum_{k_L=1}^{r_L} \prod_{\nu=1}^{d} v^{(\nu)}(x_\nu, (k_i)_{i \in S_\nu}) \prod_{\nu=d+1}^{M} v^{(\nu)}((k_i)_{i \in S_\nu})$$

where $\boldsymbol{v} = \{v^{(\nu)}\}_{\nu=1}^{M}$ is a tensor network, and each tensor $v^{(\nu)}$ is in a space $P^{(\nu)}$.

We have

$$\mathcal{M}_r = \{v = \Psi(v^{(1)}, \dots, v^{(M)}) : v^{(\nu)} \in P^{(\nu)}, 1 \le \nu \le M\},$$

where Ψ is a multilinear map.

The problem

$$\min_{v \in \mathcal{M}_r} \mathcal{J}(v)$$

can be written as an optimization problem over the parameters

$$\min_{v^{(1)}} \dots \min_{v^{(M)}} \mathcal{J}(\Psi(v^{(1)}, \dots, v^{(M)})).$$

Alternating minimization algorithm

The alternating minimization algorithm consists in solving successively minimization problems

$$\min_{v^{(\nu)} \in P^{(\nu)}} \mathcal{J}(\Psi(v^{(1)}, \dots, v^{(\nu)}, \dots, v^{(M)})) := \min_{v^{(\nu)} \in P^{(\nu)}} \mathcal{J}_\nu(v^{(\nu)}) \tag{1}$$

over the parameter $v^{(\nu)}$, letting the other parameters $v^{(\eta)}$, $\eta \neq \nu$, fixed.

When $P^{(\nu)}$ is a linear vector space, problem (1) is a linear approximation problem.

If $\mathcal{J}$ is a convex (resp. differentiable) functional, then $\mathcal{J}_\nu$ is a convex (resp. differentiable) functional.

Direct optimization in subsets of tensor networks

Other optimization algorithms (e.g. gradient descent, Newton) can be used, possibly exploiting the geometry of tree tensor networks manifolds.

Under rather standard assumptions, some results have been obtained for the convergence of algorithms: local convergence to a global optimizer, or global convergence to stationary points.

But no guaranty for obtaining a global optimizer of a general (even convex) functional in subsets of tensor networks (NP-hard problem).

Rank adaptation

For the adaptation of ranks, different strategies have been proposed:

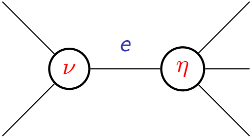
- Modified alternating minimization algorithms [Holtz et al '12] or DMRG, where rank adaptation is performed during optimization,
- Alternating minimal energy methods [Dolgov et al '14], where optimization is also combined with rank adaptation,
- Optimization in a subset with fixed rank followed by rank adaptation [Grelier/Nouy/Chevreuil'18, Grelier/Nouy/Lebrun'19,Grasedyck/Kramer '19]

Modified alternating minimization algorithm

Modified alternating minimization algorithm[1] is a modification of the alternating minimization algorithm which allows for an rank adaptation "on the fly".

It can be used for optimization with tree tensor nteworks or more general tensor networks.

At each step of the algorithm, we consider two nodes ν and η connected by an edge e and we update simultaneously the associated parameters $p^{(\nu)}$ and $p^{(\eta)}$.



[1] known as DMRG algorithm (for Density Matrix Renormalization Group) for tensor networks.

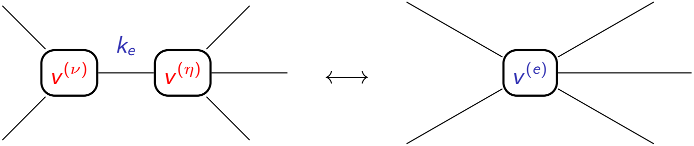
Modified alternating minimization algorithm

In the expression of a tensor $v = \Psi(v^{(1)}, \dots, v^{(M)})$, the two tensors $v^{(\nu)}$ and $v^{(\eta)}$ connected by the edge e appear as

$$\sum_{k_e=1}^{r_e} v^{(\nu)}(k_e, \dots) v^{(\eta)}(k_e, \dots) := v^{(e)}(\dots)$$

where $v^{(e)}$ is a tensor of order

$$\text{order}(v^{(e)}) = \text{order}(v^{(\nu)}) + \text{order}(v^{(\eta)}) - 2.$$



This corresponds to a new tensor networks where the nodes ν and η and edge e are replaced by a single node e, and a new parametrization

$$v = \Psi^e(\dots, v^{(e)}, \dots).$$

Modified alternating minimization algorithm

We first solve an optimization problem

$$\min_{v^{(e)}} \mathcal{J}(\Psi^e(\ldots, v^{(e)}, \ldots))$$

for obtaining an new value of the tensor $v^{(e)}$.

Then, we compute a low-rank approximation of the tensor $v^{(e)}$

$$v^{(e)}(...) \approx \sum_{k_e=1}^{r_e} v^{(\nu)}(k_e, ...)v^{(\eta)}(k_e, ...)$$

where the rank r_e in general differs from the initial rank.

In practice, the approximation is obtained using truncated singular value decomposition.

Outline

1. Higher-order singular value decomposition and tensor truncation
2. Learning from structured evaluations
3. Direct optimization in subsets of tensor networks
4. Iterative methods with tensor truncation
5. Thresholding of singular values and relaxation methods

Iterative methods with tensor truncation

Another strategy for solving an operator equation

$$Au = b$$

or a more general optimization problem

$$\min_{v \in V} \mathcal{J}(v)$$

is to rely on classical iterative methods by interpreting all standard algebraic operations on vector spaces as algebraic operations in tensor spaces.

Iterative methods with tensor truncation

As a motivating example, consider a simple Richardson algorithm

$$u^n = u^{n-1} - \omega(Au^{n-1} - b).$$

For A and b given in tensor formats, computing u^n involves standard algebraic operations.

However, the representation rank of the iterates dramatically increases since

$$\mathrm{rank}(u^n) \approx \mathrm{rank}(A)\,\mathrm{rank}(u^{n-1}) + \mathrm{rank}(u^{n-1}) + \mathrm{rank}(b).$$

This requires additional truncation steps for reducing the ranks of the iterates, such as

$$u^n = T(u^{n-1} - \omega(Au^{n-1} - b)),$$

where $T(v)$ provides a low-rank approximation of v.

We now analyze the behavior of these algorithms depending on the properties of the truncation operator T.

Fixed point iterations algorithm

Let us consider a problem which can be written as a fixed point problem

$$F(u) = u,$$

where $F : V \to V$ is a contractive map, such that for all $u, v \in V$,

$$\|F(u) - F(v)\| \leq \rho\|u - v\|,$$

with $0 \leq \rho < 1$.

Then, consider the fixed point iterations algorithm

$$u^{n+1} = F(u^n)$$

which provides a sequence $(u^n)_{n\geq 1}$ which converges to u, such that

$$\|u - u^n\| \leq \rho^n\|u - u^0\|.$$

Example

For a problem $Au = b$, consider $F(u) = u - \omega(Au - b)$, with ω such that $\|I - \omega A\| < 1$. Fixed point iterations $u^{n+1} = u^n - \omega(Au^n - b)$ correspond to Richardson iterations.

Perturbed fixed point iterations algorithm

Now consider the perturbed fixed point iterations

$$v^{n+1} = F(u^n), \quad u^{n+1} = T(v^{n+1})$$

where T is a mapping which for a tensor v provides an approximation (called truncation) $T(v)$ in a certain low-rank format $\mathcal{M}_r$.

Truncations with controlled relative precision

Suppose that the mapping T provides an approximation with relative precision ϵ, i.e.

$$\|T(v) - v\| \leq \epsilon \|v\|.$$

This is made possible by using an adaptation of the ranks.

Then the sequence $(u^n)_{n \geq 1}$ is such that

$$\|u - u^n\| \leq \gamma^n \|u - u^0\| + \frac{\epsilon}{1-\gamma} \|u\|,$$

with $\gamma = \rho(1+\epsilon)$. Therefore, if $\gamma < 1$

$$\lim\sup_{n \to \infty} \|u - u^n\| \leq \frac{\epsilon}{1-\gamma} \|u\|$$

which means that the sequence tends to enter a neighborhood of u with radius $\frac{\epsilon}{1-\gamma}\|u\|$.

The drawback of this algorithm is that the ranks of the iterates are not controlled and may become very high during the iterations.

Truncations in fixed subsets

Now consider that the mapping T provides an approximation in a fixed subset of tensors $\mathcal{M}_r$ with rank bounded by r.

Let us assume that for all v, $T(v)$ provides a quasi-optimal approximation of v such that

$$\|T(v) - v\| \leq C \min_{w \in \mathcal{M}_r} \|v - w\|. \tag{2}$$

A practical realization of a mapping T verifying (2) is provided by truncated higher-order singular value decompositions, where

$$C = O(\sqrt{d}).$$

Truncations in fixed subsets

Let u_r be an element of best approximation of u, with

$$\|u - u_r\| = \min_{v \in \mathcal{M}_r} \|u - v\|.$$

The sequence $(u^n)_{n \geq 1}$ is such that

$$\|u - u^n\| \leq \gamma^n \|u - u^0\| + \frac{C}{1-\gamma} \|u - u_r\|,$$

with $\gamma = \rho(1 + C)$. If $\gamma < 1$ (which may be quite restrictive on ρ), we obtain

$$\limsup_{n \to \infty} \|u - u^n\| \leq \frac{C}{1-\gamma} \min_{v \in \mathcal{M}_r} \|u - v\|,$$

which means that the sequence tends to enter a neighborhood of u with radius $\frac{C}{1-\gamma}\sigma_r$, where σ_r is the best approximation error of u by elements of $\mathcal{M}_r$.

An advantage of this approach is that the ranks of the iterates are controlled. A drawback is that the condition $\gamma < 1$ imposes to rely on an iterative method with small contractivity constant $\rho < (1 + C)^{-1}$, which may be quite restrictive (requires good preconditioners).

Truncations with non-expansive maps

Now we assume that the mapping T providing an approximation in low-rank format is non-expansive, i.e.

$$\|T(v) - T(w)\| \leq \|v - w\| \tag{3}$$

The sequence u^n is defined by

$$u^{n+1} = G(u^n),$$

where $G = T \circ F$ is a contractive mapping with the same contractivity constant ρ as F. Therefore, the sequence u^n converges to the unique fixed point $u^\star$ of G such that

$$G(u^\star) = u^\star,$$

with

$$\|u^\star - u^n\| \leq \rho^n \|u^\star - u^0\|.$$

The obtained approximation $u^\star$ is such that

$$(1+\rho)^{-1}\|u - T(u)\| \leq \|u - u^\star\| \leq (1-\rho)^{-1}\|u - T(u)\|.$$

A practical realization of a mapping T verifying (2) is provided by a truncation operator based on soft thresholding of singular values. The ranks of the iterates are not controlled. However, it is observed in practice that the ranks of iterates are usually lower than with truncations with controlled relative precision.

Outline

1. Higher-order singular value decomposition and tensor truncation
2. Learning from structured evaluations
3. Direct optimization in subsets of tensor networks
4. Iterative methods with tensor truncation
5. Thresholding of singular values and relaxation methods

Thresholding of singular values

Consider an order two tensor u in a Hilbert tensor space $V \otimes W$. equipped with the canonical norm.

Hard thresholding of singular values

The hard singular value thresholding operator $\mathcal{HT}_\tau$ is defined for an order-two tensor u with singular value decomposition $\sum_{k\geq 1} \sigma_k v_k \otimes w_k$ by

$$\mathcal{HT}_\tau(u) = \sum_{k\geq 1} HT_\tau(\sigma_k) v_k \otimes w_k,$$

where $HT_\tau(t) = t\, 1_{|t|>\tau}$ is the hard thresholding function such that

$$HT_\tau(\sigma_k) = \begin{cases} \sigma_k & \text{if } \sigma_k > \tau \\ 0 & \text{if } \sigma_k \leq \tau \end{cases}.$$

The error after hard thresholding is

$$\|u - \mathcal{HT}_\tau(u)\|^2 = \sum_{k\geq 1} \sigma_k^2\, 1_{\sigma_k \leq \tau}.$$

$\mathcal{HT}_\tau(u)$ is a solution of the problem

$$\min_v \|u - v\|^2 + \tau^2 \operatorname{rank}(v)$$

where $\operatorname{rank}(v) = \|\sigma(v)\|_0$.

Soft thresholding of singular values

The soft singular value thresholding operator $\mathcal{ST}_\tau$ is defined for a tensor u with singular value decomposition $\sum_{k\geq 1} \sigma_k v_k \otimes w_k$ by

$$\mathcal{ST}_\tau(u) = \sum_{k\geq 1} ST_\tau(\sigma_k) v_k \otimes w_k,$$

where $ST_\tau(t) = (|t| - \tau)_+ \operatorname{sign}(t)$ is the soft thresholding function, such that

$$ST_\tau(\sigma_k) = (\sigma_k - \tau)_+ = \begin{cases} \sigma_k - \tau & \text{if } \sigma_k \geq \tau \\ 0 & \text{if } \sigma_k < \tau \end{cases}.$$

The error after soft thresholding is

$$\|u - \mathcal{ST}_\tau(u)\|^2 = \sum_{k\geq 1} (\sigma_k - (\sigma_k - \tau)_+)^2 = \sum_{\sigma_k \leq \tau} {\sigma_k}^2 + \sum_{\sigma_k > \tau} \tau^2.$$

Soft thresholding of singular values

$\mathcal{ST}_\tau(u)$ is a solution of the problem

$$\min_v \frac{1}{2}\|u - v\|^2 + \tau\|\sigma(v)\|_1$$

where $\|\sigma(v)\|_1$ is the nuclear norm of v, which is a convex regularization of the functional $v \mapsto \text{rank}(v)$.

In convex analysis, $\mathcal{ST}_\tau$ is known as the proximal operator of the convex function $v \mapsto \tau\|\sigma(v)\|_1$.

The operator $\mathcal{ST}_\tau$ is non-expansive, that means for all u, v,

$$\|\mathcal{ST}_\tau(u) - \mathcal{ST}_\tau(v)\| \leq \|u - v\|,$$

which is an important property for the analysis of algorithms with tensor truncations.

Convex relaxation

A general optimization problem over a subset of tensors with bounded rank

$$\min_{\mathrm{rank}(v) \leq r} \mathcal{J}(v)$$

is equivalent to

$$\min_{v} \mathcal{J}(v) + \tau \, \mathrm{rank}(v)$$

for some value of τ.

A convex optimization problem is obtained by replacing $\mathrm{rank}(v) = \|\sigma(v)\|_0$ by the function $\|\sigma(v)\|_1 = \|v\|_*$ (the nuclear norm of v)

$$\min_{v} \mathcal{J}(v) + \tau \|v\|_*$$

Proximal algorithms

Consider the problem

$$\min_{v} \mathcal{J}(v) + \tau\|v\|_*$$

A proximal algorithm constructs a sequence $(u^n)_{n\geq 1}$ as follows.

At iteration n, we linearize the function $\mathcal{J}$ around u^n and define u^{n+1} as the solution of

$$\min_{v} \mathcal{J}(u^n) + (\nabla\mathcal{J}(u^n), v - u^n) + \frac{\beta}{2}\|u - u^n\|^2 + \tau\|v\|_*$$

where β is a parameter.

This is equivalent to solving

$$\min_{v} \frac{1}{2}\|v - (u^n - \beta^{-1}\nabla\mathcal{J}(u^n))\|^2 + \frac{\tau}{\beta}\|v\|_*$$

whose solution is provided by

$$u^{n+1} = \mathsf{ST}_{\tau/\beta}(u^n - \beta^{-1}\nabla\mathcal{J}(u^n))$$

where $\mathsf{ST}_{\tau/\beta}$ is the proximal operator of $v \mapsto \frac{\tau}{\beta}\|v\|_*$.

Hard and soft singular values thresholding for higher order tensors

For a higher order tensor u in a Hilbert tensor space $V = V_1 \otimes \ldots \otimes V_d$, we can naturally define hard and soft singular values thresholding operators $\mathcal{HS}_\tau^\alpha$ and $\mathcal{ST}_\tau^\alpha$ associated with the singular value decomposition of the matricisation $\mathcal{M}_\alpha(u)$ of u.

These operators are such that

$$\mathcal{HS}_\tau^\alpha(u) = \arg\min_v \|u - v\|^2 + \tau^2 \operatorname{rank}_\alpha(v),$$

and

$$\mathcal{ST}_\tau^\alpha(u) = \arg\min_v \frac{1}{2}\|u - v\|^2 + \tau\|\sigma^\alpha(u)\|_1.$$

Hard and soft singular values thresholding for higher order tensors

Hard and soft thresholding operators can then be defined for the approximation in a tree-based format $\mathcal{T}_r^T(V)$, with T a dimension tree (or a subset T of a dimension tree),

Hard and soft thresholding operators $\mathcal{HT}_\tau^T$ and $\mathcal{ST}_\tau^T$ can be respectively defined as compositions of hard and soft thresholding operators (sequence of truncations from the root to the leaves),

$$\mathcal{HT}_\tau^T = \mathcal{HT}_\tau^{\alpha_M} \circ \ldots \circ \mathcal{HT}_\tau^{\alpha_1}$$

and

$$\mathcal{ST}_\tau^T = \mathcal{ST}_\tau^{\alpha_M} \circ \ldots \circ \mathcal{ST}_\tau^{\alpha_1}$$

where the set of nodes $\{\alpha_1, \ldots, \alpha_M\} = T \setminus \{D\}$ is sorted by increasing level.

The soft-thresholding operator $\mathcal{ST}_\tau^T$ is non-expansive, i.e.

$$\|\mathcal{ST}_\tau^T(u) - \mathcal{ST}_\tau^T(v)\| \leq \|u - v\|$$

for all tensors u, v.

See [Rauhut'17] and [Bachmayr'16] for further details and applications to tensor completion and solution of operator equations.

Convex relaxation for tree-based formats

Given a tree-based format $\mathcal{T}_r^T(V)$, a convex relaxation of the problem

$$\min_{v \in \mathcal{T}_r^T(V)} \mathcal{J}(v)$$

can be defined as

$$\min_{v \in V} \mathcal{J}(v) + \tau \sum_{\alpha \in T \setminus \{D\}} \|\sigma^{\alpha}(u)\|_1. \qquad (\star)$$

- Algorithms based on soft thresholding of singular values appear as specific algorithms for solving the relaxed optimization problem $(\star)$.

Convex relaxation for tree-based formats

Given a tree-based format $\mathcal{T}_r^T(V)$, a convex relaxation of the problem

$$\min_{v \in \mathcal{T}_r^T(V)} \mathcal{J}(v)$$

can be defined as

$$\min_{v \in V} \mathcal{J}(v) + \tau \sum_{\alpha \in T \setminus \{D\}} \|\sigma^\alpha(u)\|_1. \qquad (\star)$$

- Algorithms based on soft thresholding of singular values appear as specific algorithms for solving the relaxed optimization problem $(\star)$.
- But this relaxation is known to be far from optimal convex relaxation.

Convex relaxation for tree-based formats

Given a tree-based format $\mathcal{T}_r^T(V)$, a convex relaxation of the problem

$$\min_{v \in \mathcal{T}_r^T(V)} \mathcal{J}(v)$$

can be defined as

$$\min_{v \in V} \mathcal{J}(v) + \tau \sum_{\alpha \in T \setminus \{D\}} \|\sigma^\alpha(u)\|_1. \qquad (\star)$$

- Algorithms based on soft thresholding of singular values appear as specific algorithms for solving the relaxed optimization problem $(\star)$.
- But this relaxation is known to be far from optimal convex relaxation.
- For Tucker tensors, a better convex relaxation is based on tensor nuclear norm [Yuan/Zhang'16].

Convex relaxation for tree-based formats

Given a tree-based format $\mathcal{T}_r^T(V)$, a convex relaxation of the problem

$$\min_{v \in \mathcal{T}_r^T(V)} \mathcal{J}(v)$$

can be defined as

$$\min_{v \in V} \mathcal{J}(v) + \tau \sum_{\alpha \in T \setminus \{D\}} \|\sigma^{\alpha}(u)\|_1. \qquad (\star)$$

- Algorithms based on soft thresholding of singular values appear as specific algorithms for solving the relaxed optimization problem ($\star$).
- But this relaxation is known to be far from optimal convex relaxation.
- For Tucker tensors, a better convex relaxation is based on tensor nuclear norm [Yuan/Zhang'16].
- Finding a good convex relaxation for general tree-based formats remains an open problem.

References I

- **Higher-order singular value decompositions and related tensor truncation schemes**

L. De Lathauwer, B. De Moor, and J. Vandewalle.
A multilinear singular value decomposition.
SIAM J. Matrix Anal. Appl., 21(4):1253–1278, 2000.

Ivan V Oseledets and Eugene E Tyrtyshnikov.
Breaking the curse of dimensionality, or how to use svd in many dimensions.
SIAM Journal on Scientific Computing, 31(5):3744–3759, 2009.

L. Grasedyck.
Hierarchical singular value decomposition of tensors.
SIAM J. Matrix Anal. Appl., 31:2029–2054, 2010.

M. Che and Y. Wei.
Randomized algorithms for the approximations of tucker and the tensor train decompositions.
Advances in Computational Mathematics, 45(1):395–428, 2019.

Y. Sun, Y. Guo, C. Luo, J. Tropp, and M. Udell.
Low-rank tucker approximation of a tensor from streaming data.
SIAM Journal on Mathematics of Data Science, 2(4):1123–1150, 2020.

B. Huber, R. Schneider, and S. Wolf.
A randomized tensor train singular value decomposition.
In *Compressed sensing and its applications*, pages 261–290. Springer, 2017.

References II

V. Ehrlacher, L. Grigori, D. Lombardi, and H. Song.
Adaptive hierarchical subtensor partitioning for tensor compression.
SIAM Journal on Scientific Computing, 43(1):A139–A163, 2021.

L. Grigori and S. Kumar.
Parallel tensor train through hierarchical decomposition.
2020.

H. A. Daas, G. Ballard, and P. Benner.
Parallel algorithms for tensor train arithmetic.
arXiv preprint arXiv:2011.06532, 2020.

- **Relaxation methods**

M. Yuan and C.-H. Zhang.
On tensor completion via nuclear norm minimization.
Foundations of Computational Mathematics, 16(4):1031–1068, 2016.

H. Rauhut, R. Schneider, and Z. Stojanac.
Tensor Completion in Hierarchical Tensor Representations, pages 419–450.
Springer International Publishing, Cham, 2015.

H. Rauhut, R. Schneider, and Z. Stojanac.
Low rank tensor recovery via iterative hard thresholding.
Linear Algebra and its Applications, 523:220–262, 2017.

References III

M. Bachmayr and R. Schneider.
Iterative methods based on soft thresholding of hierarchical tensors.
Foundations of Computational Mathematics, pages 1–47, 2016.

- **Approximation with structured sampling**

J. Ballani, L. Grasedyck, and M. Kluge.
Black box approximation of tensors in hierarchical tucker format.
Linear Algebra and its Applications, 438(2):639 – 657, 2013.
Tensors and Multilinear Algebra.

I. Oseledets and E. Tyrtyshnikov.
TT-cross approximation for multidimensional arrays.
Linear Algebra And Its Applications, 432(1):70–88, JAN 1 2010.

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

C. Haberstich, A. Nouy, and G. Perrin.
Active learning of tree tensor networks using optimal least-squares.
arXiv preprint arXiv:2104.13436, 2021.

- **Low-rank methods for tensor-structured equations: surveys**

References IV

B. B. Khoromskij and C. Schwab.
Tensor-structured Galerkin approximation of parametric and stochastic elliptic pdes.
SIAM Journal on Scientific Computing, 33(1):364–385, 2011.

B. Khoromskij.
Tensors-structured numerical methods in scientific computing: Survey on recent advances.
Chemometrics and Intelligent Laboratory Systems, 110(1):1 – 19, 2012.

A. Nouy.
Low-rank methods for high-dimensional approximation and model order reduction.
ArXiv e-prints, November 2015.

Markus Bachmayr, Reinhold Schneider, and André Uschmajew.
Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations.
Foundations of Computational Mathematics, pages 1–50, 2016.

- **Low-rank methods for tensor-structured equations: truncated iterations**

D. Kressner and C. Tobler.
Low-rank tensor Krylov subspace methods for parametrized linear systems.
SIAM Journal on Matrix Analysis and Applications, 32(4):1288–1316, 2011.

J. Ballani and L. Grasedyck.
A projection method to solve linear systems in tensor format.
Numerical Linear Algebra with Applications, 20(1):27–43, 2013.

References V

M. Bachmayr and W. Dahmen.
Adaptive near-optimal rank tensor approximation for high-dimensional operator equations.
Foundations of Computational Mathematics, 15(4):839–898, 2015.

Markus Bachmayr and Reinhold Schneider.
Iterative methods based on soft thresholding of hierarchical tensors.
Foundations of Computational Mathematics, pages 1–47, 2016.

- **Alternating minimization and rank adaptation**

S. Holtz, T. Rohwedder, and R. Schneider.
The alternating linear scheme for tensor optimization in the tensor train format.
SIAM Journal on Scientific Computing, 34(2):A683–A713, 2012.

S. V. Dolgov and D. V. Savostyanov.
Alternating minimal energy methods for linear systems in higher dimensions.
SIAM Journal on Scientific Computing, 36(5):A2248–A2271, 2014.

- **Greedy algorithms**

E. Cances, V. Ehrlacher, and T. Lelievre.
Convergence of a greedy algorithm for high-dimensional convex nonlinear problems.
Mathematical Models & Methods In Applied Sciences, 21(12):2433–2467, December 2011.

A. Falcó and A. Nouy.
Proper generalized decomposition for nonlinear convex problems in tensor banach spaces.
Numerische Mathematik, 121:503–530, 2012.

References VI

- **Preconditioners in low-rank formats**

  B. Khoromskij.
  Tensor-structured preconditioners and approximate inverse of elliptic operators in $\mathbb{R}^d$.
  Constructive Approximation, 30(3):599–620, 2009.

  L. Giraldi, A. Nouy, and G. Legrain.
  Low-rank approximate inverse for preconditioning tensor-structured linear systems.
  SIAM Journal on Scientific Computing, 36(4):A1850–A1870, 2014.

  M. Bachmayr and V. Kazeev.
  Stability of low-rank tensor representations and structured multilevel preconditioning for elliptic pdes.
  Foundations of Computational Mathematics, 20(5):1175–1236, 2020.

- **Learning with tensor networks**

  L. Grasedyck and S. Krämer.
  Stable ALS approximation in the TT-format for rank-adaptive tensor completion.
  Numerische Mathematik, 143(4):855–904, 2019.

  E. Stoudenmire and D. J. Schwab.
  Supervised learning with tensor networks.
  In *Advances in Neural Information Processing Systems*, pages 4799–4807, 2016.

References VII

E. Grelier, A. Nouy, and M. Chevreuil.
Learning with tree-based tensor formats.
arXiv e-prints, page arXiv:1811.04455, Nov. 2018.

E. Grelier, A. Nouy, and R. Lebrun.
Learning high-dimensional probability distributions using tree tensor networks.
arXiv preprint arXiv:1912.07913, 2019.

B. Michel and A. Nouy.
Learning with tree tensor networks: complexity estimates and model selection. To appear in Bernoulli.
arXiv e-prints, page arXiv:2007.01165, July 2020.

- **Geometry of tensor manifolds and geometrical approaches**

P-A Absil, Robert Mahony, and Rodolphe Sepulchre.
Optimization algorithms on matrix manifolds.
Princeton University Press, 2009.

S. Holtz, T. Rohwedder, and R. Schneider.
On manifolds of tensors of fixed TT-rank.
Numerische Mathematik, 120(4):701–731, 2012.

A. Uschmajew and B. Vandereycken.
Geometric Methods on Low-Rank Matrix and Tensor Manifolds, pages 261–313.
Springer International Publishing, Cham, 2020.

References VIII

M. Billaud-Friess, A. Falco, and A. Nouy.
Principal bundle structure of matrix manifolds.
ArXiv e-prints, May 2017.

A. Falcó, W. Hackbusch, and A. Nouy.
On the Dirac–Frenkel variational principle on tensor Banach spaces.
Foundations of Computational Mathematics, 19(1):159–204, Feb 2019.

A. Falco, W. Hackbusch, and A. Nouy.
Geometry of tree-based tensor formats in tensor Banach spaces, 2021.

- **Dynamical low-rank methods for evolution problems**

C. Lubich, T. Rohwedder, R. Schneider, and B. Vandereycken.
Dynamical approximation by hierarchical tucker and tensor-train tensors.
SIAM Journal on Matrix Analysis and Applications, 34(2):470–494, 2013.

E. Musharbash.
Dynamical low rank approximation of pdes with random parameters.
page 194, 2017.

M. Bachmayr, H. Eisenmann, E. Kieri, and A. Uschmajew.
Existence of dynamical low-rank approximations to parabolic problems, 2020.

M. Billaud-Friess, A. Falcó, and A. Nouy.
A new splitting algorithm for dynamical low-rank approximation motivated by the fibre bundle structure of matrix manifolds, to appear in BIT Numerical Mathematics 2021.