Nick Vannieuwenhoven
with B. Sprangers
KU LEUVEN

Group-invariant tensor train networks for supervised learning

Overview

Overview

1. Introduction

A key task in supervised machine learning is determining a good **model** from a class of **parameterized models**.[1] That is, find the element from

$$\mathcal{M} = \{f_\alpha : \mathbb{R}^m \to \mathbb{R}^n \mid \alpha \in \mathbb{R}^p\}$$

that maximizes some **performance criterion** p, given suitable amounts of **data** $\mathbf{d} \in \mathbb{R}^m \times \mathbb{R}^n$:

$$\max_{f_\alpha \in \mathcal{M}} p(f_\alpha, \mathbf{d}).$$

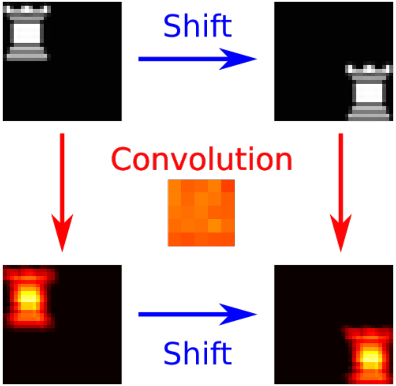
Examples of model classes include:

1. linear, quadratic, logistic, whateveryoulikebest regression
2. graphical models
3. neural networks

and the performance criterion could be maximizing prediction accuracy on a set of data.

[1] Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006

In recent years, researchers in machine learning recognized that convolutional neural networks contain a powerful **inductive bias**; their outputs are invariant under translations:



2

So mathematically, if S is a shifting map, and f is the convolution map, then we have

$$f(S\mathbf{x}) = Sf(\mathbf{x})$$

That is, the diagram on the left commutes. (Since both maps are linear, this states that S and f are commuting matrices.)

[2] Figure from Kayhan (2020). URL: https://medium.com/@oskyhn77789/current-convolutional-neural-networks-are-not-translation-equivariant-2f04bb9062e3

This realization led to the desire to incorporate additional constraints **explicitly** into machine learning models $f : \mathbb{R}^m \to \mathbb{R}^n$, such as

1. **invariance**: $f(A\mathbf{x}) = f(\mathbf{x})$, and
2. **equivariance**: $f(A\mathbf{x}) = Bf(\mathbf{x})$.

Herein A and B are usually linear maps. See Cohen[3] for an introduction to this area.

Many **data augmentation** strategies also try to realize these properties **implicitly**. These come with an additional computational cost, as each training example could be subjected to multiple transformations A.

[3] Cohen, *Equivariant convolutional networks*, PhD thesis, 2021.

In this talk, we focus on simple **tensor-based models** for supervised learning.[4,5]

Here, the input data is first passed through a **feature map**

$$\Phi : \mathbb{R}^m \to \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \cdots \times \mathbb{R}^{n_k},$$

as in a **kernel method**, and then supplied as input to a **multilinear map**

$$f : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \cdots \times \mathbb{R}^{n_k} \to \mathbb{R}^{n_{k+1}}$$

By imposing suitable structures on f, this map can be computed efficiently, comprising its **kernel trick**.

[4] Stoudenmire, Schwab, *Supervised Learning with Tensor Networks*, 30th Conference on Neural Information Processing Systems, 2016.

[5] Novikov, Trofimov, Oseledets, *Exponential machines*, Bull. Polish Acad. Sci.: Tech. Sci., 2018.

Central question

How can we **parameterize** the **multilinear maps**

$$f : V_1 \times V_2 \times \cdots \times V_k \to V_{k+1}$$

invariant under the action of chosen invertible linear maps $M_g^i : V_i \to V_i$:

$$f(M_g^1 \mathbf{x}_1, M_g^2 \mathbf{x}_2, \ldots, M_g^k \mathbf{x}_k) = M_g^{k+1} f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k), \quad g = 1, \ldots, s,$$

for all $\mathbf{x}_i \in V_i$.

Most of the work presented stems from our desire to understand prior work by Finzi, Welling and Wilson.[6]

[6]Finzi, Welling, Wilson, *A practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups*, Proceedings of the 38th International Conference on Machine Learning, 2021.

Overview

2. Group-invariant tensors: Tensors

Let us first recall how multilinear maps could be parameterized.

Recall that a **multilinear function** is a map

$$f : V_1 \times V_2 \times \cdots \times V_k \to V_{k+1}$$

from the vector spaces V_i to the vector space V_{k+1} which is linear in each of its arguments:

$$\begin{aligned} &f(\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \alpha \cdot \mathbf{x}_i + \beta \cdot \mathbf{y}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_k) \\ &\qquad = \alpha \cdot f(\mathbf{x}_1, \ldots, \mathbf{x}_k) + \beta \cdot f(\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{y}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_k), \qquad i = 1, \ldots, k. \end{aligned}$$

Because of the universal property of the tensor product,[7] there is **a bijection between multilinear functions** f **and order-**$(k+1)$ **tensors**

$$\mathcal{F} \in (V_1^* \otimes V_2^* \otimes \cdots \otimes V_k^*) \otimes V_{k+1}.$$

The notation V_i^* means the **dual space** of V_i, i.e., the linear space of functions from V_i to the base field ($\mathbb{R}$).

[7] Greub, *Multilinear Algebra*, Springer, 1978.

By exploiting the multilinearity of f and the tensor product $\otimes$, it can be shown that

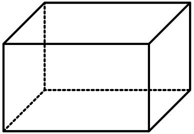
$$\mathcal{F} = \sum_{j_1}^{n_1} \cdots \sum_{j_d=1}^{n_k} \mathbf{e}_{j_1}^{1*} \otimes \cdots \otimes \mathbf{e}_{j_k}^{k*} \otimes f(\mathbf{e}_{j_1}^{1}, \ldots, \mathbf{e}_{j_k}^{k}),$$

where $\mathbf{e}_1^i, \ldots, \mathbf{e}_{n_i}^i$ forms an orthonormal basis of the n_i-dimensional vector space V_i, and $\mathbf{e}_j^{*i}$ denotes the dual basis vector of $\mathbf{e}_j^i$; that is,

$$\mathbf{e}_j^{*i}(\mathbf{e}_{j'}^i) = \delta_{jj'} = \begin{cases} 1 & j = j' \\ 0 & \text{otherwise} \end{cases}.$$

(This is completely analogous to how a matrix represents a linear map.)

One can represent this tensor $\mathcal{F}$ by a $(k+1)$-**array**

$\mathcal{F} \simeq$  $\in \mathbb{R}^{n_1 \times \cdots \times n_k \times n_{k+1}}$.

You can think of $\mathcal{F}$ as containing the function evaluation $f(\mathbf{e}^1_{i_1}, \ldots, \mathbf{e}^k_{i_k})$ at position $(i_1, \ldots, i_k)$ in this array.

Hence, the **unconstrained model space** of multilinear functions can be parameterized as

$$\mathcal{M} = \{f : V_1 \times \cdots \times V_k \to V_{k+1} \text{ multilinear}\} \simeq \mathbb{R}^{n_1 \times \cdots \times n_k \times n_{k+1}} \simeq \mathbb{R}^{n_1 \cdots n_{k+1}}.$$

2. Group-invariant tensors: Invariant multilinear maps

For neural networks, Cohen and Welling[8] introduced the concept of G-**invariance** for groups G. In our setting, we want to consider the maps f that satisfy

$$\forall \mathbf{x}_i \in V_i : \quad f(M_g^1 \mathbf{x}_1, \dots, M_g^k \mathbf{x}_k) = M_g^{k+1} f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k), \quad g = 1, \dots, s,$$

for the fixed tuples, which we call **invariance relations**,

$$M_g := (M_g^1, \dots, M_g^{k+1}) \in \mathrm{Aut}(V_1) \times \cdots \times \mathrm{Aut}(V_{k+1})$$

for $g = 1, \dots, s$.

Recall that $\mathrm{Aut}(V_i)$ is the subspace of bijective linear maps from V_i into itself.

[8] Cohen, Welling, *Group equivariant convolutional networks*, Proceedings of the 33rd International Conference on Machine Learning (ICML), 2016.

Lemma

Let G denote the set of **all** *invariance relations satisfied by f subject to the previously imposed invariance relations. Then, $(G, \circ)$ is a group.*

□

Lemma

Let G denote the set of **all** *invariance relations satisfied by f subject to the previously imposed invariance relations. Then, $(G, \circ)$ is a group.*

Indeed, if $M_g, M_h \in G$, then

$$f(M_h^1 M_g^1 \mathbf{x}_1, \ldots, M_h^k M_g^k \mathbf{x}_k) = M_h^{k+1} f(M_g^1 \mathbf{x}_1, \ldots, M_g^k \mathbf{x}_k) = M_h^{k+1} M_g^{k+1} f(\mathbf{x}_1, \ldots, \mathbf{x}_k),$$

so that $M_h \circ M_g := (M_h^1 M_g^1, \ldots, M_h^{k+1} M_g^{k+1}) \in G$.

□

Lemma

Let G denote the set of **all** *invariance relations satisfied by f subject to the previously imposed invariance relations. Then, $(G, \circ)$ is a group.*

Indeed, if $M_g, M_h \in G$, then

$$f(M_h^1 M_g^1 \mathbf{x}_1, \ldots, M_h^k M_g^k \mathbf{x}_k) = M_h^{k+1} f(M_g^1 \mathbf{x}_1, \ldots, M_g^k \mathbf{x}_k) = M_h^{k+1} M_g^{k+1} f(\mathbf{x}_1, \ldots, \mathbf{x}_k),$$

so that $M_h \circ M_g := (M_h^1 M_g^1, \ldots, M_h^{k+1} M_g^{k+1}) \in G$.

Moreover, we have for all $M_g \in G$ that

$$M_g^{k+1} f((M_g^1)^{-1}\mathbf{x}_1, \ldots, (M_g^k)^{-1}\mathbf{x}_k) = f(M_g^1 (M_g^1)^{-1}\mathbf{x}_1, \ldots, M_g^k (M_g^k)^{-1}\mathbf{x}_k) = f(\mathbf{x}_1, \ldots, \mathbf{x}_k),$$

which implies that $M_g^{-1} := ((M_g^1)^{-1}, \ldots, (M_g^{k+1})^{-1}) \in G$. □

Moreover, it is immediately verified that the projection maps

$$\pi_i : G \to \mathrm{Aut}(V_i), \quad (M^1, \ldots, M^{k+1}) \mapsto M^i$$

are **group homomorphisms**. That is,

1. $\pi_i((\mathrm{Id}_{V_1}, \ldots, \mathrm{Id}_{V_{k+1}})) = \mathrm{Id}_{V_i}$, and
2. $\pi_i(M_h \circ M_g) = M_h^i M_g^i = \pi_i(M_h)\pi_i(M_g)$.

A map $\rho : G \to \mathrm{Aut}(V)$ that maps an abstract group G homomorphically into the group of automorphisms on a vector space V is called a **group representation of G on V**.

All of the foregoing entails that multilinear maps satisfying

$$f(M_g^1\mathbf{x}_1, M_g^2\mathbf{x}_2, \ldots, M_g^k\mathbf{x}_k) = M_g^{k+1} f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k), \quad g = 1, \ldots, s,$$

are G-**invariant** (with representations $\rho_i : G \to V_i$) for the group $G = \langle M_1, \ldots, M_s \rangle$.[9] That is, the above equation holds for all $M_g \in G$, not only for the generators.

Note that conversely one could start from an abstract group along with suitable representations and impose G-invariance in this way on f.

[9] Lang, *Algebra*, 3rd ed., Springer, 2002.

2. Group-invariant tensors: Invariant tensors

We saw that a natural way of imposing invariance relations on a multilinear map f leads to f's G-invariance. What does this entail for the associated tensor $\mathcal{F}$?

Assume we have linear maps $U_i^* : V_i^* \to V_i^*$, $i = 1, \dots, k$, and $U_{k+1} : V_{k+1} \to V_{k+1}$. Let

$$\mathcal{F} = \sum_{j_1=1}^{n_1} \cdots \sum_{j_{k+1}=1}^{n_{k+1}} \mathcal{F}_{j_1,\dots,j_{k+1}} \mathbf{e}_{j_1}^{1*} \otimes \cdots \otimes \mathbf{e}_{j_k}^{k*} \otimes \mathbf{e}_{j_{k+1}}^{k+1}.$$

Then, the **multilinear multiplication** of these maps with $\mathcal{F}$ is defined as[10]

$$(U_1^* \otimes \cdots \otimes U_k^* \otimes U_{k+1})(\mathcal{F}) := \sum_{j_1=1}^{n_1} \cdots \sum_{j_{k+1}=1}^{n_{k+1}} \mathcal{F}_{j_1,\dots,j_{k+1}} (U_1^* \mathbf{e}_{j_1}^{1*}) \otimes \cdots \otimes (U_k^* \mathbf{e}_{j_k}^{k*}) \otimes (U_{k+1} \mathbf{e}_{j_{k+1}}^{k+1}).$$

[10] Greub, *Multilinear Algebra*, Springer, 1978

Proposition (Sprangers, Vannieuwenhoven, 2022)

Let $f : V_1 \times \cdots \times V_k \to V_{k+1}$ *be a multilinear map,* $\mathcal{F} \in V_1^* \otimes \cdots \otimes V_k^* \otimes V_{k+1}$ *the associated tensor,* $G = \langle g_1, \ldots, g_s \rangle$ *a finitely-generated group, and* $\rho_i : G \to \mathrm{Aut}(V_i)$ *representations. Then,* f *is* G*-invariant if and only if*

$$\mathcal{F} = (\rho_1^*(g) \otimes \cdots \otimes \rho_k^*(g) \otimes \rho_{k+1}(g))(\mathcal{F}), \quad \forall g \in \{g_1, \ldots, g_s\},$$

where $\rho^*(g) = \big(\rho(g)\big)^{-\top}$ *is the dual representation.*

Note that the inversion and transposition make sense in the dual representation[11] because if $f : V \to W$ then

$$f^{-1} : W \to V, \quad f^\top : W^* \to V^*, \quad \text{so } f^{-\top} : V^* \to W^*$$

[11] Lang, *Algebra*, 3rd ed., Springer, 2002.

Example

Consider the case of a **linear map** $\mathcal{F} : V \to W$ that satisfies the following equality

$$\forall \mathbf{v} \in V : \quad L\mathcal{F}\mathbf{v} = \mathcal{F}M\mathbf{v}$$

for $L \in \mathrm{Aut}(W)$ and $M \in \mathrm{Aut}(V)$.

As this holds for all $\mathbf{v}$, we have equality of linear maps: $L\mathcal{F} = \mathcal{F}M$. Hence, equivalently,

$$L\mathcal{F}M^{-1} = \mathcal{F}.$$

Vectorizing, this is equivalent to

$$(M^{-\top} \otimes L)(\mathrm{vec}(\mathcal{F})) = \mathrm{vec}(\mathcal{F}),$$

having used a standard property of the Kronecker product.

Overview

3. Efficiently constructing group-invariant tensors

Inspecting the definition of G-invariant tensors, we see that for all $g \in G$:

$$\begin{aligned}\mathcal{F} &= (\rho_1^*(g), \dots, \rho_k^*(g), \rho_{k+1}(g), \dots, \rho_d(g)) \cdot \mathcal{F} \\ &:= (\rho_1^{-\top}(g) \otimes \cdots \otimes \rho_k^{-\top}(g) \otimes \rho_{k+1}(g) \otimes \cdots \otimes \rho_d(g))(\mathcal{F}),\end{aligned}$$

where $\otimes$ can be interpreted as the Kronecker product.

This is an interesting **simultaneous eigenvector problem** in which $\mathcal{F}$ is the common eigenvector corresponding to eigenvalue 1 of the tensor-structured matrices

$$\rho_1^{-\top}(g) \otimes \cdots \otimes \rho_k^{-\top}(g) \otimes \rho_{k+1}(g) \otimes \cdots \otimes \rho_d(g), \quad g \in G$$

Corollary

The G-invariant tensors form a linear subspace of $V_1^* \otimes \cdots \otimes V_k^* \otimes V_{k+1} \otimes \cdots \otimes V_d$.

In the remainder, we consider orthogonal group representations. The results can be extended to **normal group representations** as well.

Orthogonal representation

Let h be an inner product on V. A representation is **orthogonal** if $\rho(g) : V \to V$ is an isometry $\forall g \in G$.

For orthogonal representations $\rho^{-\top}(g) = \rho(g)$, so we can simplify the notation.

Returning to our eigenvalue problem, we have

$$\rho_1(g) \otimes \cdots \otimes \rho_d(g) = (U^1 \otimes \cdots \otimes U^d)(\Lambda^1 \otimes \cdots \otimes \Lambda^d)(U^1 \otimes \cdots \otimes U^d)^H$$

where U^i is a unitary matrix and Λ^i is a diagonal matrix containing the (complex) eigenvalues such that

$$\rho_i(g) = U^i \Lambda^i (U^i)^H.$$

Let

$$U^1_\star \odot \cdots \odot U^d_\star = [\mathbf{u}^1_{i_1} \otimes \cdots \otimes \mathbf{u}^d_{i_d}]_{i_1,\ldots,i_d}$$

be the matrix of eigenvectors corresponding to eigenvalue 1, i.e.,

$$\Lambda^1_{i_1,i_1} \cdots \Lambda^d_{i_d,i_d} = 1.$$

The space of G-invariant tensors is a subspace of $U_\star^1 \odot \cdots \odot U_\star^d$, so that

$$\mathcal{F} = (U_\star^1 \odot \cdots \odot U_\star^d)\mathbf{v}$$

for some $\mathbf{v}$. Plugging this into our eigenvalue problem, we get

$$\mathbf{v} = \underbrace{(U_\star^1 \odot \cdots \odot U_\star^d)^H}_{U_\star^H} \underbrace{(\rho_1(g) \otimes \cdots \otimes \rho_d(g))}_{B_g} \underbrace{(U_\star^1 \odot \cdots \odot U_\star^d)}_{U_\star} \mathbf{v}, \quad \forall g \in G.$$

This tensor-structured matrix can be computed efficiently.

Our key result is that the projected simultaneous eigenproblem can be reduced to a single eigenproblem. This result can be viewed as a version of the **first projection formula** in representation theory more suitable for tensor product representations.[12]

Proposition (Sprangers, Vannieuwenhoven, 2022)

Let $B_g \in \mathbb{C}^{m\times m}$ be unitary matrices whose rightmost eigenvalue is real. Let $U_\star \in \mathbb{C}^{m\times n}$ be a matrix with orthogonal columns, i.e., $U_\star^H U_\star = I_n$. Let $A_g = U_\star^H B_g U_\star$. Then,

$$A_1\mathbf{v} = \lambda_1\mathbf{v}, \ \ldots, \ A_s\mathbf{v} = \lambda_s\mathbf{v}$$

if and only if $\frac{1}{s}(\frac{1}{\lambda_1}A_1 + \cdots + \frac{1}{\lambda_s}A_s)\mathbf{v} = \mathbf{v}$.

[12] Fulton, Harris, *Representation Theory: A First Course*, Springer, 2004.

With these ingredients, we propose the following algorithm.

Input: Normal representation matrices B_i^k of $\rho_k(g_i)$ for $G = \langle g_0, g_1, \ldots, g_s \rangle$.

1. Compute for $k = 1, \ldots, d$ the small-scale eigendecompositions

$$\rho_k(g_0) = B_0^k = U^k \Lambda^k (U^k)^H.$$

2. Find all indices $(i_1, \ldots, i_d)$ such that $\Lambda^1_{i_1,i_1} \cdots \Lambda^d_{i_d,i_d} = 1$ and set $U_\star^k = [\mathbf{u}^k_{i_k}]_{i_k}$.
3. Compute

$$A = \frac{1}{s} \sum_{i=1}^{s} \left((U_\star^1)^H B_i^1 U_\star^1 \right) \circledast \cdots \circledast \left((U_\star^d)^H B_i^d U_\star^d \right).$$

4. Compute a Schur decomposition $A = QTQ^H$, where T is upper triangular, and extract the eigenspace Q corresponding to eigenvalue 1.

Output: The orthonormal basis $(U_\star^1 \odot \cdots \odot U_\star^d)Q$.

Overview

4. Group-invariant tensor train networks

Recall our supervised learning setup where we compose

$$\Phi : \mathbb{R}^m \to \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_k} \simeq \mathbb{R}^{n_1+\cdots+n_{k+1}} \quad \text{and} \quad f : \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_k} \to \mathbb{R}^{n_{k+1}}$$

where Φ was called a **kernel map** and f is multilinear. Mathematically, this is equivalent to

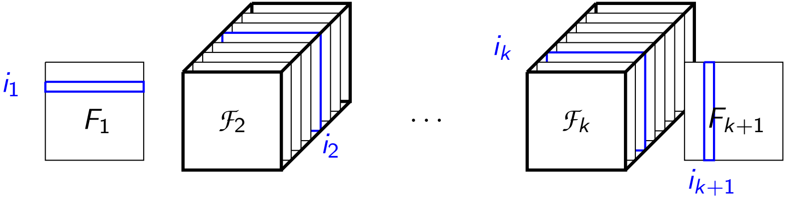
$$f \circ \Phi = \mathcal{F} \circ \otimes \circ \Phi,$$

where $\mathcal{F}$ is the tensor representing f. This emphasizes that $\otimes \circ \Phi$ maps into a high-dimensional space. And $\mathcal{F} \in (\mathbb{R}^{n_1 \times \cdots \times n_k})^* \otimes \mathbb{R}^{n_{k+1}}$ is a **linear map** $\mathcal{F} : \mathbb{R}^{n_1 \times \cdots \times n_k} \to \mathbb{R}^{n_{k+1}}$.

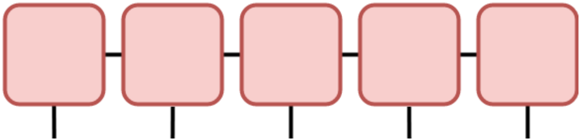
This has all the hallmarks of a **kernel method**. Except: we need a **kernel trick** because applying a linear map to vectors in $\mathbb{R}^{n_1 \times \cdots \times n_k}$ is too costly (in memory and time)!

We say that $\mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_{k+1}}$ admits a **tensor trains decomposition**[13] with **bond dimensions** $(r_1, \ldots, r_k)$ if each entry of the tensor is a contracted **matrix chain multiplication**, like so

$$f_{i_1, i_2, \ldots, i_k, i_{k+1}} =$$

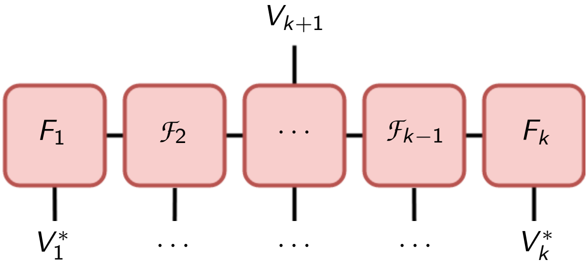


Graphically, the above is represented as



[13] Fannes, Nachtergaele, Werner, Comm. Math. Phys. 144, pp. 443–490, 1992.

In the case where $\mathcal{F} \in (V_1^* \otimes \cdots \otimes V_k^*) \otimes V_{k+1}$ represents a multilinear function with one output vector space V_{k+1}, we have



Note that you can play with the location of the output vector space V_{k+1}.

To impose G-invariance on multilinear maps that correspond to tensor trains decompositions with small bond dimensions no new theoretical developments are needed.
Singh, Pfeifer, and Vidal[14] namely proved the following result.

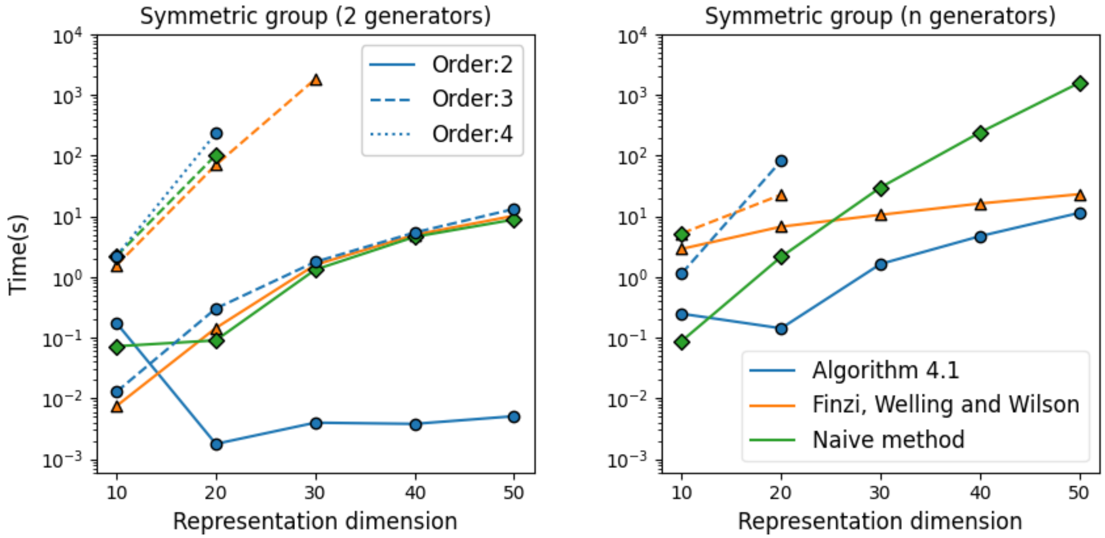
Proposition (Singh, Pfeifer, Vidal, 2010)

There exists a tensor trains decomposition with minimal bond dimensions of a G-invariant tensor in which all core tensors are themselves G-invariant.

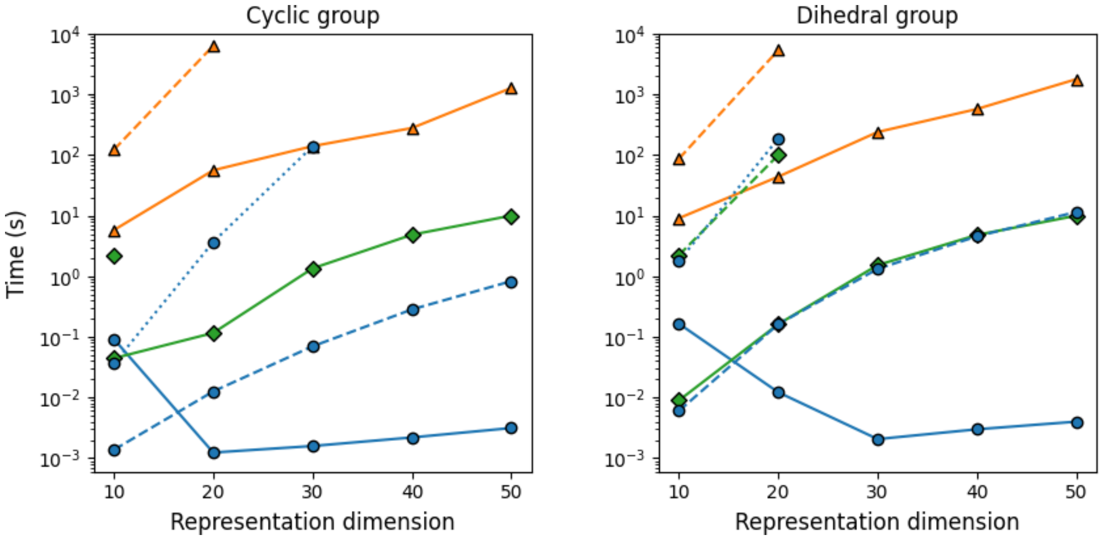
[14] Singh, Pfeifer, Vidal, Phys. Rev. A: At. Mol. Opt. Phys., 82, 2010.

Overview

5. Experimental results: Basis construction performance



5. Experimental results: Basis construction performance



5. Experimental results: G-invariant networks for transcription prediction

We applied group-invariant tensor train networks to a supervised learning task: Binary prediction whether a **transcription factor (protein) will bind to a DNA sequence.**

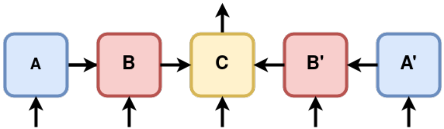
A data set with 3 transcription factors (MAX, CTCF, SPI1) was curated by Zhou, Shrikumar and Kundaje,[15] along with $10,000$ DNA strands per transcription factor. The dataset is already partitioned $40\%/30\%/30\%$ into a training, test, and validation set.

DNA strands are **reverse complement symmetric** (Zhou, Shrikumar, Kundaje, 2020):

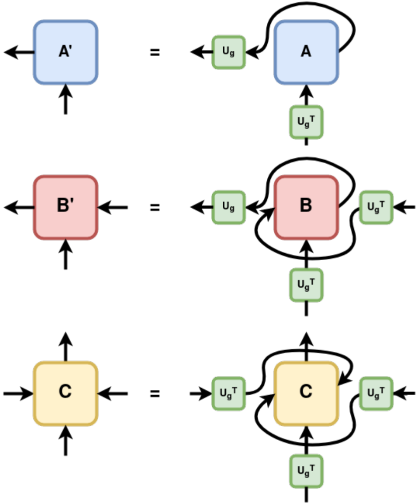
- **Complement invariance** arises from the nucleobase pairings in the double helix (A $\leftrightarrow$ T, and G $\leftrightarrow$ C).
- **Reverse invariance** occurs because if a transcription factor binds to a DNA strand, then it also binds on the same strand rotated by π radians by rotating the protein likewise.

[15] Zhou, Shrikumar, Kundaje, Benchmarking reverse-complement strategies for deep learning models in genomics, bioRxiv:2020.11.04.368803, 2020.

The tensor trains network has 1001 cores with output in the middle. All cores have bond dimension b.
The nucleobases are one-hot encoded as a length-4 binary vector.



The complement invariance can be modeled as the group $G(G,*) = (\mathbb{Z}_2, +_2)$



The training setup was as follows:

- 100 epochs with batch size of 100,
- binary cross-entropy loss and 2-regularization on the variational parameters,
- softmax activation function at output node,
- stochastic gradient descent with Nesterov momentum with a fraction of 0.2

The optimal hyperparameters (found by non-exhaustive manual experimentation) vary depending on the prediction task:

Task	Bond dimension	Regularization	Epochs	Learning rate
MAX	3	0.005	100	0.001
CTCF	8	0.005	100	0.01
SPI1	8	0.003	100	0.01

Average* results over 5 runs of our model together with the results from the state-of-the-art convolutional neural network introduced by Mallet and Vert[16], which in addition to reverse complement symmetry also takes into account a translation invariance, are as follows:

Dataset	**Model**	**AUROC**	**Standard deviation**
CTCF	Ours	94.10%	0.21%
	Benchmark	**98.84%**	0.056%
SPI1	Ours	96.53%	0.030%
	Bechmark	**99.26%**	0.034%
MAX	Ours	**97.06%**	0.011%
	Benchmark	92.80%	0.26%

[16] Mallet, Vert, *Reverse-Complement Equivariant Networks for DNA Sequences*, NeurIPS, 2021.

Overview

6. Conclusions

Invariance relationships are naturally modeled with groups, leading to the concept of group-invariant tensor train networks. A new algorithm was proposed for constructing a basis of G-invariant tensors, outperforming the state of the art by several orders of magnitude.

For more details, see:

B. Sprangers and N. Vannieuwenhoven,
Group-invariant tensor train networks for supervised learning,
arXiv:2206.15051, 2022.