MAX PLANCK INSTITUTE FOR DYNAMICS OF COMPLEX TECHNICAL SYSTEMS MAGDEBURG

COMPUTATIONAL METHODS IN SYSTEMS AND CONTROL THEORY

BiG max MAX PLANCK RESEARCH NETWORK on big-data-driven materials science

IMPRS ProEng International Max Planck Research School for Advanced Methods in Process and Systems Engineering Magdeburg

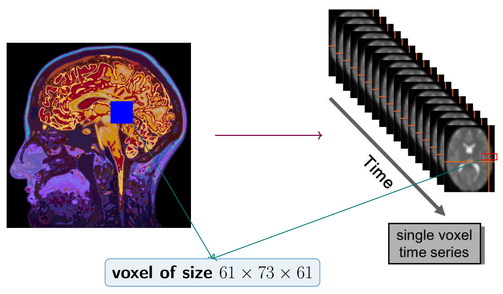
# Efficient Structure-preserving Support Tensor Train Machine


Kirandeep Kour[1], Sergey Dolgov[2], Martin Stoll[3], Peter Benner[1,3]

[1]Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany
[2]University of Bath, Bath, UK
[3]Technische Universität, Chemnitz, Germany


## Motivation

An example: classification of **ADHD** infected brain image and normal brain image:




Dataset
+ Multi Dimensional (MD)
+ Small Sample Size (SSS)

Binary Classification
+ Learning Model
+ Nonlinear Boundary + MD + SSS

Efficienct Model
+ Robustness
+ State-of-the-art for MD + SSS

## Tensor-Train (TT) Decomposition

- A tensor is a multidimensional array.
- Matricization is unfolding of a tensor along the $M^{\text{th}}$-mode for an $M-$dimensional tensor.
- Manipulating a tensor is often prone to the curse of dimensionality $\mathcal{O}(\mathcal{I}^M)$.

$$x_{i_1 i_2 \dots i_M} \cong \sum_{r_0,\dots,r_M} \mathcal{G}^{(1)}_{r_0,i_1,r_1} \mathcal{G}^{(2)}_{r_1,i_1,r_2} \cdots \mathcal{G}^{(M)}_{r_{M-1},i_M,r_M},$$
$$\mathcal{X} \cong \langle\langle \mathcal{G}^{(1)}, \mathcal{G}^{(2)}, \dots, \mathcal{G}^{(M)} \rangle\rangle,$$

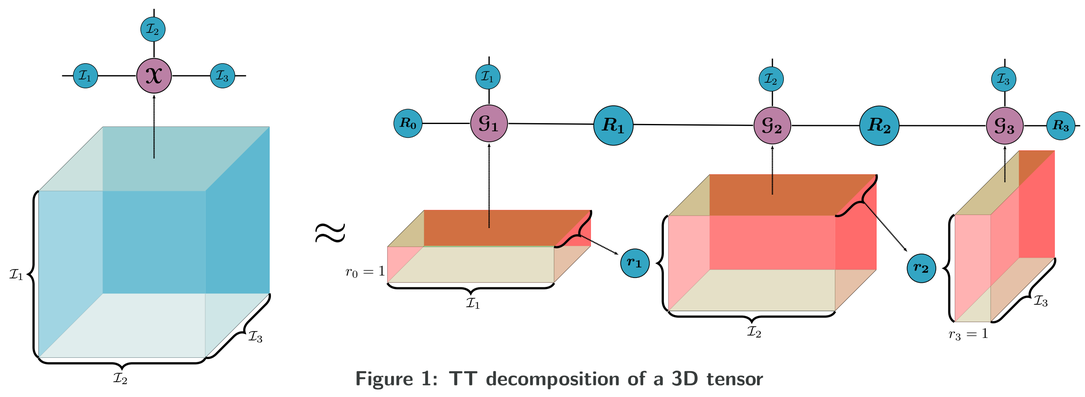



**Figure 1: TT decomposition of a 3D tensor**

- A tensor decomposition works efficiently avoiding the curse of dimensionality.
- Specially, TT decomposition provides a compact form with storage demands in $\mathcal{O}(MIR^2)$.
- TT decomposition adequately exploits low-rank structures.

## Support Tensor Machine: Binary Classification

Model for Small Sample Size:

- Binary Classification → $x$ - input (all the data points), $y$ - output $\{-1, 1\}$
- Learning Model → Support Vector Machine (SVM)
- Nonlinear Boundary → Projection into higher-dimensional space

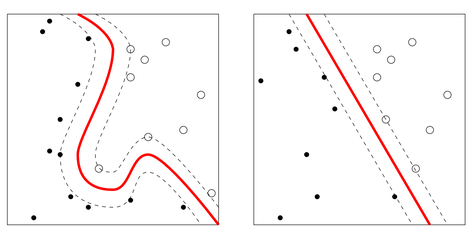


**Figure 2: Non-linear(kernel) and linear SVM**

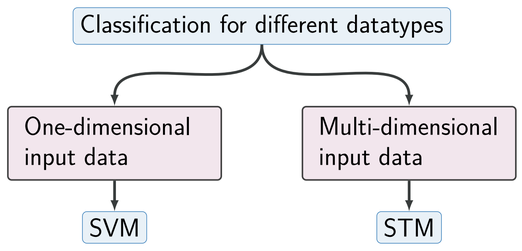



**Figure 3: SVM vs. STM**

Linear Boundary Optimization for SVM (Dual):

$$\max_{\alpha_1,\dots,\alpha_N} \left( \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \right)$$
$$\text{subject to} \quad 0 \le \alpha_i \le C, \quad \sum_{i=1}^{N} \alpha_i y_i = 0.$$

Dual optimization problem corrsponding to input data tensor for Support Tensor Machine (STM) is as follows:

$$\max_{\alpha_1,\dots,\alpha_N} \left( \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \mathcal{X}_i, \mathcal{X}_j \rangle \right)$$
$$\text{subject to} \quad 0 \le \alpha_i \le C, \quad \sum_{i=1}^{N} \alpha_i y_i = 0.$$

## Nonlinear Boundary

### Projection into higher-dimensional space

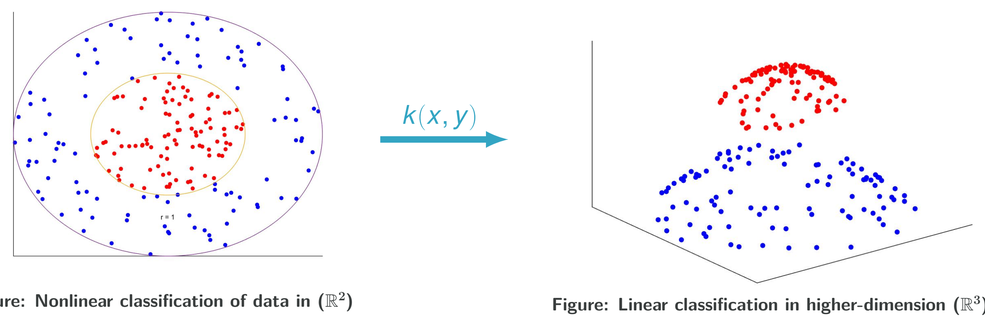



**Figure: Nonlinear classification of data in ($\mathbb{R}^2$)**

**Figure: Linear classification in higher-dimension ($\mathbb{R}^3$)**

The Feature map from lower dimensional nonlinear boundary value problem to linear high dimensional problem,

$$\phi : \mathcal{X}(\text{input}) \longrightarrow \mathcal{V}(\text{feature space})$$

Using **Representation theorem**

$$\langle \phi(x), \phi(x') \rangle = k(x, x')$$

Here, $k$ is called kernel function, $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. We use the *Gaussian kernel*

$$k(x, x') = \exp\left( - \frac{\|x_i - x'_j\|^2}{2\sigma^2} \right)$$

For STM $k(x, x') \to k(\mathcal{X}, \mathcal{X}')$, Computationally expensive term.

## Multi-Way Nonlinear Mapping

A pictorial representation of a nonlinear mapping:

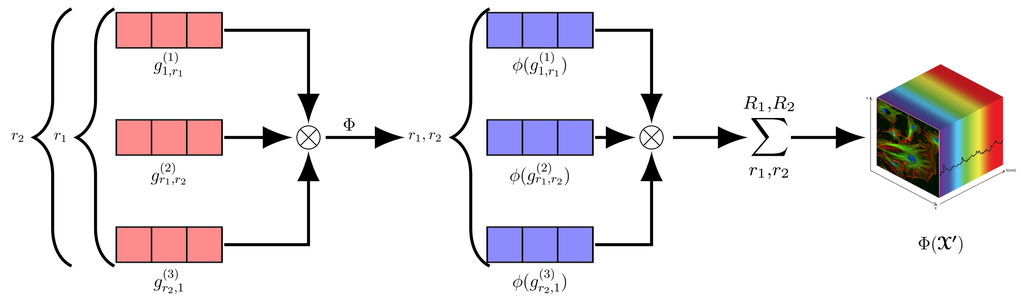



**Figure 4: Multiway nonlinear mapping of TT vectors**

$$\langle \Psi(\mathcal{X}), \Psi(\mathcal{Y}) \rangle = k(\mathcal{X}, \mathcal{Y}) = \sum_{r_1,r_2,q_1,q_2}^{R_1,R_2,Q_1,Q_2} k(g^{(1)}_{1,r_1}, p^{(1)}_{1,q_1}) k(g^{(2)}_{r_1,r_2}, p^{(2)}_{q_1,q_2}) k(g^{(3)}_{r_2,1}, p^{(3)}_{q_2,1})$$

## Building Efficient Model

- Using TT decomposition
  - Stable algorithm
  - Depends on matrix SVD
  - Leads to over-fitting
  - Blocks might not be unique
- Using CP decomposition
  - Simplicity
  - Finding best rank: NP-hard

### TT-CP Expansion

Given $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, $x_{i_1,i_2,i_3} \approx \sum_{r_1,r_2=1}^{R_1,R_2} \mathcal{G}^{(1)}_{i_1,r_1} \mathcal{G}^{(2)}_{r_1,i_2,r_2} \mathcal{G}^{(3)}_{r_2,i_3}$

CP approximation $x_{i_1,i_2,i_3} \approx \sum_{r=1}^{R_1 R_2} H^{(1)}_{i_1,r} H^{(2)}_{i_2,r} H^{(3)}_{i_3,r}$,

$r_1, r_2 \to r = r_1 + (r_2 - 1) R_1$

$H^{(1)}_{i_1,r} = \mathcal{G}^{(1)}_{i_1,r_1} \mathbf{1}_{r_2}, \quad H^{(3)}_{i_3,r} = \mathbf{1}_{r_1} \mathcal{G}^{(3)}_{r_2,i_3}$

### Stability: Uniqueness of SVD (UoSVD)

UoSVD ensures that "close" tensor produce "close" TT-cores. Also, makes model stable w.r.t. different rank truncation.

$$\mathcal{X}_{(1)} = \sigma_1 u_1 v_1^\top + \dots + \sigma_{I_1} u_{I_1} v_{I_1}^\top,$$
$$\bar{u}_{r_1} := u_{r_1} / \text{sign}(u_{r_1}(i^*_{r_1})),$$
$$\bar{v}_{r_1} := v_{r_1} \cdot \text{sign}(u_{r_1}(i^*_{r_1})),$$
$$i^*_{r_1} = \arg\max_{i=1,\dots,I_1} |u_{r_1}(i)|.$$

### State-of-the-art: Norm Equilibration

It makes a significant increment in the classification accuracy. For a given rank-$r$ TT-CP decomposition $[\![H^{(1)}, H^{(2)}, H^{(3)}]\!]$,

$$\|\mathcal{N}_r\| = \left\|H^{(1)}_r\right\| * \left\|H^{(2)}_r\right\| * \left\|H^{(3)}_r\right\|,$$
$$H^{(k)}_r := \frac{H^{(k)}_r}{\left\|H^{(k)}_r\right\|} * \|\mathcal{N}_r\|^{1/3}, \qquad k = 1, 2, 3.$$

$$\langle \Psi(\mathcal{X}), \Psi(\mathcal{Y}) \rangle = k(\mathcal{X}, \mathcal{Y}) = \sum_{i,j=1}^{R} k(h^{(1)}_i, q^{(1)}_j) k(h^{(2)}_i, q^{(2)}_j) k(h^{(3)}_i, q^{(3)}_j)$$

## Numerical Results

**Experiment Setup**:

- Using TT-Toolbox for TT decomposition,
- Using LIBSVM library for SVM.
- k-fold cross validation for splitting training and testing data

**Parameter Tuning**:

- TT-ranks $R \in \{1, 2, \dots 10\}$
- Gaussian Kernel $\sigma \in \{2^{-8}, 2^{-7}, \dots, 2^7, 2^8\}$
- Trade-off parameter in SVM $C \in \{2^{-8}, 2^{-7}, \dots, 2^7, 2^8\}$

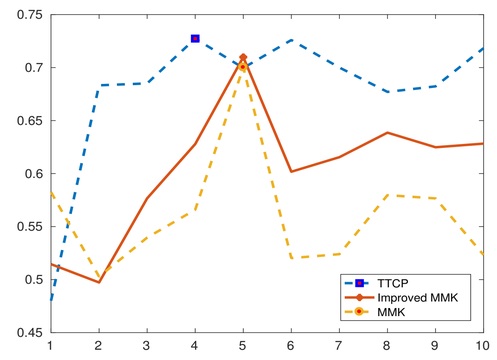



**Figure 5:** Accuracy for ADNI dataset w.r.t. truncated TT-CP rank

**Datasets**:

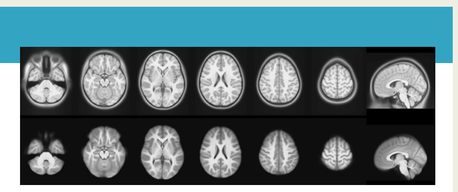
- Alzheimer Disease (ADNI[1])
- Attention Deficit Hyperactivity Disorder (ADHD[2])

**Table 1:** Best average classification accuracy for different methods and datasets

| Methods | ADNI | ADHD |
|---|---|---|
| SVM | 49 % | 52% |
| STuM | 51 % | 54% |
| DuSK | 55 % | 58% |
| MMK | 69 % | 60% |
| Improved MMK | 71 % | 61% |
| TT-MMK | **73%** | **64%** |

http://adni.loni.usc.edu/
http://neurobureau.projects.nitrc.org/ADHD200/Data.html

## Conclusion and Outlook



Conclusion

- Low rank-method "Exact TT-CP Expansion"
- Kernel Model for less data with higher dimensions
- Approximating kernel not only reduced cost, also increases accuracy
- Stability and computational cost

Toolboxes and Codes:

- TT-Toolbox by Ivan Oseledets and Sergey Dolgov: `https://github.com/oseledets/TT-Toolbox`
- LIBSVM by Chih-J. Lin: `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`
- `https://arxiv.org/abs/2002.05079`

## Selected References


[1] L. He, X. Kong, P. S. Yu, X. Yang, A. B. Ragin, and Z. Hao, *Dusk: A dual structure-preserving kernel for supervised tensor learning with applications to neuroimages*, in Proc. of the 2014 SIAM Internat. Conference on Data Mining, SIAM, 2014, pp. 127–135.

[2] L. He, C.-T. Lu, H. Ding, S. Wang, L. Shen, P. S. Yu, and A. B. Ragin, *Multi-way multi-level kernel modeling for neuroimaging classification*, in Proc. of the IEEE Conference Computer Vision and Pattern Recognition, 2017, pp. 356–364.

[3] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317.