Supervised Learning and Canonical Decomposition of Multivariate Functions

N.D. Sidiropoulos (joint work with N. Kargas)
The Supervised Learning Problem

Categorical (classification, binary or FA)
Real-valued (prediction, regression)
Complex-valued (channel; MRI k-space)
AKA: I/O (Nonlinear) System Identification

Categorical (classification, binary or FA)
Real-valued (prediction, regression)
Complex-valued (channel; MRI k-space)
(Deep) Neural Networks

- Most popular method for learning to mimic nonlinear functions
- Some theory ... but, for most part ...
  - Don’t understand why they work so well
  - Choosing architecture is art
  - Hard to interpret
- Against all odds and principles!
(Deep) Neural Networks

- Most popular method for learning to mimic nonlinear functions
- Some theory ... but, for most part ...
  - Don’t understand why they work so well
  - Choosing architecture is art
  - Hard to interpret
- Against all odds and principles!

- This talk: principled alternative
- Based on tensor principal components
- Advantages: `universal', intuitive, interpretable, backed by theory
- Works with incomplete input data – important in practice
Introduction

- **General nonlinear function identification**
  - ‘Supervised’ - from input-output data
  - Function approximation problem
  - Identifiability? Performance? Complexity?

- **Applications**
  - Machine learning
  - Dynamical system identification and control
  - Communications
Motivation

- Course grade prediction
- Drug response prediction
Motivation

Text classification

Channel estimation
Sneak preview

- **Deep neural networks**
  - Work very well in practice
  - Hard to interpret
  - Difficult to tune

- **In this work:**
  - Simple and elegant alternative
  - Low-rank tensor decomposition
  - Model any nonlinearity
  - Identification guarantees
Canonical Polyadic Decomposition (CPD)

- An N-way tensor (multi-way array) admits a decomposition of rank $F$ it can be decomposed as a sum of $F$ rank-1 tensors
  \[ \mathcal{X} = \sum_{f=1}^{F} \mathbf{a}_f^1 \odot \mathbf{a}_f^2 \odot \cdots \odot \mathbf{a}_f^N \]
- Tensor rank is smallest $F$ for which such decomposition exists \(\rightarrow\) Canonical

- Element-wise: $\mathcal{X}(i_1, \ldots, i_N) = \sum_{f=1}^{F} \prod_{n=1}^{N} a_{f}^n(i_n)$
- Matrix unfolding: $\mathcal{X}^{(n)} = (\mathbf{A}_N \odot \cdots \odot \mathbf{A}_{n+1} \odot \mathbf{A}_{n-1} \cdots \odot \cdots \mathbf{A}_1)\mathbf{A}_n^T$
- Vector: $\text{vec}(\mathcal{X}) = (\mathbf{A}_N \odot \cdots \odot \mathbf{A}_1)\mathbf{1}$
Prior work

- Tensor modeling of low-order multivariate polynomial systems (Rendle, 2010)
- A multivariate polynomial of order $d$ is represented by a tensor of order $d$

\[
y = w^T x \\
y = x^T W x \\
y = W_{\times_1} x_{\times_2} x_{\times_3} x
\]
Prior work

- Number of parameters grows exponentially with the order $d$
  - Assume that the coefficient tensor is low-rank

- **Drawbacks**
  - Require prior knowledge of polynomial order
  - Assuming polynomial of a given degree can be restrictive
  - Simplest rank=1 model → number of parameters grows linearly with $d$
  - Cannot model high-degree polynomial functions
We propose:

- Single high-order tensor for learning a general nonlinear system

Canonical System Identification (CSID)

- **Claims:**
  - CPD can model *any* nonlinearity (even of $\infty$ order) for high-enough rank. Even for low ranks, it can model highly nonlinear operators.
  - Provably correct nonlinear system identification from limited samples, when the tensor is low rank.
  - Even when not low rank identification of the principal components!
Rank of generic nonlinear systems?

- **Seperable function:** \( y = f(x_1, \ldots, x_N) = \prod_{n=1}^{N} f_n(x_n) \)
  - Rank: 1
  - e.g., \( f(x_1, \ldots, x_N) = \prod_{n=1}^{N} \text{sign}(x_n) \)

- **Sum of separable functions:** \( y = f(x_1, \ldots, x_N) = \sum_{n=1}^{N} f_n(x_n) \)
  - Maximal rank: N
  - e.g., \( f(x_1, \ldots, x_N) = \sum_{n=1}^{N} \text{sign}(x_n) \)

- **Sum of pairwise functions:** \( y = f(x_1, \ldots, x_N) = \sum_{i=1}^{N} \sum_{j>i} f_{ij}(x_i, x_j) \)
  - Maximal rank: \( \frac{IN^2}{2} \ll I^{N-1} \)

- **Other nonlinear systems?**
Problem formulation

- Each input vector $[x_m(1), \ldots, x_m(N)]$ is viewed as a cell multi-index and the cell content is the estimated response of the system:

$$\min_{\mathcal{X}} \frac{1}{M} \sum_{m=1}^{M} (y_m - \mathcal{X}(x_m(1), \ldots, x_m(N)))^2$$

- We aim for the principal components of the nonlinear operator:

$$\min_{\mathcal{X}, \{A_n\}_{n=1}^{N}} \frac{1}{M} \sum_{m=1}^{M} (y_m - \mathcal{X}(x_m(1), \ldots, x_m(N)))^2 + \sum_{n=1}^{N} \rho \|A_n\|_{F}^2$$

subject to

$$\mathcal{X} = \sum_{f=1}^{F} A_1(:, f) \odot \cdots \odot A_N(:, f)$$
Datasets often contain both categorical and ordinal predictors.

\[
\min_{\mathcal{X}, \{A_n\}_{n=1}^N} \frac{1}{M} \left\| \sqrt{\mathcal{W}} \odot (\mathcal{Y} - \mathcal{X}) \right\|_F^2 + \sum_{n=1}^N \rho \|A_n\|_F^2 + \sum_{n=1}^N \mu_n \|T_n A_n\|_F^2
\]

subject to \( \mathcal{X} = \sum_{f=1}^F A_1(:, f) \odot \cdots \odot A_N(:, f) \),

where

\[
T_n = \begin{bmatrix}
1 & -1 \\
1 & -1 \\
\vdots & \vdots \\
1 & -1
\end{bmatrix} \quad \text{or} \quad T_n = \begin{bmatrix}
1 & -2 & 1 \\
1 & -2 & 1 \\
\vdots & \vdots & \vdots \\
1 & -2 & 1
\end{bmatrix}
\]
Tensor completion: Identifiability

- **Probabilistic results**
  - Adaptive sampling (Krishnamurthy and Singh 2013)
  - Random sampling with orthogonal factors (Jain and Oh 2014)
  - Random sampling assuming low mode-n ranks (Huang et al. 2014)

- **Deterministic results**
  - Fiber sampling (Sorensen and De Lathauwer 2019)
  - Regular sampling (Kanatsoulis et al. 2019)
Tensor completion: Identifiability

- Depends on how the x-samples are generated – randomly or systematically, and if randomly from what distribution.

- Practical experience: generic sample complexity for randomly drawn point samples ~ degrees of freedom $O(FNI)$ in the model. Proven for randomly drawn linear (generalized, aggregated) samples in

- ... but not (yet?) for point samples.

- For $F < I$, can show that for uniform random point samples, the sample complexity for our low-rank model is $O(\sqrt{FIN} \log(N))$, using
Algorithm

- **Alternating minimization**
  - Exploit sparsity (Smith and Karypis 2015)
  - Cyclically update variables
  - Lightweight row-wise updates

\[
\min_{\mathcal{X}, \{A_n\}_{n=1}^N} \frac{1}{M} \left\| \sqrt{W} \otimes (Y - \mathcal{X}) \right\|_F^2 + \sum_{n=1}^N \rho \|A_n\|_F^2 + \sum_{n=1}^N \mu_n \|T_n A_n\|_F^2
\]

subject to \( \mathcal{X} = \sum_{f=1}^F A_1(:,f) \odot \cdots \odot A_N(:,f) \),

- **Large scale problems** SGD, Block-stochastic GD
Let $\mathcal{O}$ and $\mathcal{M}$ denote the indices of the observed and missing entries of a single observation

$$f(x_{\mathcal{O}}) = \mathbb{E}_{x_{\mathcal{M}}|x_{\mathcal{O}}} [f(x_{\mathcal{O}}, x_{\mathcal{M}})] = \sum_{x_{\mathcal{M}}} P_{X_{\mathcal{M}}|X_{\mathcal{O}}} (x_{\mathcal{M}}|x_{\mathcal{O}}) f(x_{\mathcal{O}}, x_{\mathcal{M}})$$

We adopt a simple rank-1 joint PMF model estimated via the empirical one-dimensional marginal distributions (K. Huang, N. D. Sidiropoulos, 2017)

$$f(x_{\mathcal{O}}) = \mathbb{E}_{x_{\mathcal{M}}|x_{\mathcal{O}}} [f(x_{\mathcal{O}}, x_{\mathcal{M}})] = \mathcal{X}(i_1, \ldots, i_T, \ldots, :) \times_{T+1} \mathbf{p}_{T+1} \cdots \times_{T+L} \mathbf{p}_{N}$$

$$= \sum_{f=1}^{F} \prod_{n=1}^{T} \mathbf{A}_n(i_n, f) \prod_{n=T+1}^{N} \mathbf{p}_n^T \mathbf{A}_n(:, f)$$

Multi-output regression

- No correlation between the K output variables → build K independent models
- Output variables are usually correlated

**Better approach:**
- Build a single model capable of predicting all K outputs \( \mathcal{X} = [\mathbf{A}_1, \ldots, \mathbf{A}_N, \mathbf{V}]_F \)
- The new tensor model can be described by N+1 factors
- No modification is needed for the ALS updates
- Prediction: \( \mathcal{X}(i_1, \ldots, i_N,:) = (\mathbf{A}_1(i_1,: \otimes \cdots \otimes \mathbf{A}_N(i_N,:)) \mathbf{V}^T \)

Experiments

- Regression task using 9 UCI datasets
- Grade prediction task
  - 20 CS courses selected from University of Minnesota
  - 20 independent models using 34 courses as predictors

- 10 Monte Carlo simulations
- 80% training, 20% test (5-fold cross-validation for parameter selection)
- Evaluate the performance using RMSE
## Dataset information

<table>
<thead>
<tr>
<th>Dataset</th>
<th>N</th>
<th>M</th>
<th>Type</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concrete Compressive Strength</td>
<td>8</td>
<td>1030</td>
<td>Ordinal</td>
<td>$y \in (2, 83)$</td>
</tr>
<tr>
<td>SkillCraft Master Table</td>
<td>18</td>
<td>3337</td>
<td>Ordinal</td>
<td>$y \in (1, 7)$</td>
</tr>
<tr>
<td>Abalone</td>
<td>8</td>
<td>4177</td>
<td>Mixed</td>
<td>$y \in (1, 29)$</td>
</tr>
<tr>
<td>Wine Quality</td>
<td>11</td>
<td>4898</td>
<td>Ordinal</td>
<td>$y \in (3, 9)$</td>
</tr>
<tr>
<td>Combined Cycle Power Plant</td>
<td>4</td>
<td>9568</td>
<td>Ordinal</td>
<td>$y \in (420, 496)$</td>
</tr>
<tr>
<td>Physicochemical Properties</td>
<td>9</td>
<td>45730</td>
<td>Ordinal</td>
<td>$y \in (0, 21)$</td>
</tr>
<tr>
<td>Energy efficiency (2)</td>
<td>8</td>
<td>788</td>
<td>Ordinal</td>
<td>$y_1 \in (6, 44)$</td>
</tr>
<tr>
<td>Parkinsons Telemonitoring (2)</td>
<td>19</td>
<td>5875</td>
<td>Mixed</td>
<td>$y_1 \in (5, 40)$</td>
</tr>
<tr>
<td>Bike Sharing (2)</td>
<td>12</td>
<td>17379</td>
<td>Mixed</td>
<td>$y_1 \in (0, 367)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>N</th>
<th>M</th>
<th>Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI-1</td>
<td>34</td>
<td>996</td>
<td>0.54</td>
</tr>
<tr>
<td>CSCI-2</td>
<td>34</td>
<td>990</td>
<td>0.55</td>
</tr>
<tr>
<td>CSCI-3</td>
<td>34</td>
<td>983</td>
<td>0.55</td>
</tr>
<tr>
<td>CSCI-4</td>
<td>34</td>
<td>958</td>
<td>0.55</td>
</tr>
<tr>
<td>CSCI-5</td>
<td>34</td>
<td>953</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-6</td>
<td>34</td>
<td>931</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-7</td>
<td>34</td>
<td>911</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-8</td>
<td>34</td>
<td>898</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-9</td>
<td>34</td>
<td>867</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-10</td>
<td>34</td>
<td>856</td>
<td>0.57</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>N</th>
<th>M</th>
<th>Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI-11</td>
<td>34</td>
<td>704</td>
<td>0.57</td>
</tr>
<tr>
<td>CSCI-12</td>
<td>34</td>
<td>696</td>
<td>0.58</td>
</tr>
<tr>
<td>CSCI-13</td>
<td>34</td>
<td>650</td>
<td>0.57</td>
</tr>
<tr>
<td>CSCI-14</td>
<td>34</td>
<td>636</td>
<td>0.59</td>
</tr>
<tr>
<td>CSCI-15</td>
<td>34</td>
<td>600</td>
<td>0.57</td>
</tr>
<tr>
<td>CSCI-16</td>
<td>34</td>
<td>598</td>
<td>0.57</td>
</tr>
<tr>
<td>CSCI-17</td>
<td>34</td>
<td>529</td>
<td>0.56</td>
</tr>
<tr>
<td>CSCI-18</td>
<td>34</td>
<td>519</td>
<td>0.55</td>
</tr>
<tr>
<td>CSCI-19</td>
<td>34</td>
<td>431</td>
<td>0.55</td>
</tr>
<tr>
<td>CSCI-20</td>
<td>34</td>
<td>403</td>
<td>0.55</td>
</tr>
</tbody>
</table>
Results: Full data

- Baselines: Ridge Regression (RR), Support Vector Regression (SVR), Decision Tree (DT), Neural network: multilayer perceptron (MLP).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RR</th>
<th>SVR (RBF)</th>
<th>SVR (polynomial)</th>
<th>DT</th>
<th>MLP (5 Layer)</th>
<th>CSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy Eff. (1)</td>
<td>2.91±0.17</td>
<td>2.68±0.17</td>
<td>4.09±0.49</td>
<td>0.56±0.03</td>
<td>0.48±0.06 [50]</td>
<td>0.39±0.05</td>
</tr>
<tr>
<td>Energy Eff. (2)</td>
<td>3.09±0.19</td>
<td>3.03±0.21</td>
<td>4.14±0.44</td>
<td>1.86±0.19</td>
<td>0.97±0.14 [50]</td>
<td>0.57±0.09</td>
</tr>
<tr>
<td>C. Comp. Strength</td>
<td>10.47±0.42</td>
<td>9.72±0.38</td>
<td>11.30±0.36</td>
<td>6.57±0.82</td>
<td>4.92±0.63 [50]</td>
<td>4.67±0.50</td>
</tr>
<tr>
<td>SkillCraft Master Table</td>
<td>1.68±1.61</td>
<td>0.99±0.03</td>
<td>1.22±0.05</td>
<td>1.03±0.04</td>
<td>1.00±0.03 [10]</td>
<td>0.91±0.02</td>
</tr>
<tr>
<td>Abalone</td>
<td>2.25±0.10</td>
<td>2.19±0.08</td>
<td>3.90±3.43</td>
<td>2.35±0.08</td>
<td>2.09±0.09 [10]</td>
<td>2.23±0.09</td>
</tr>
<tr>
<td>Wine Quality</td>
<td>0.76±0.02</td>
<td>0.69±0.02</td>
<td>1.01±0.39</td>
<td>0.75±0.03</td>
<td>0.72±0.02 [10]</td>
<td>0.70±0.02</td>
</tr>
<tr>
<td>Parkinsons Tel. (1)</td>
<td>7.51±0.11</td>
<td>6.66±0.14</td>
<td>7.89±0.88</td>
<td>2.40±0.26</td>
<td>3.60±0.18 [100]</td>
<td>1.33±0.10</td>
</tr>
<tr>
<td>Parkinsons Tel. (2)</td>
<td>9.75±0.15</td>
<td>9.14±0.17</td>
<td>10.04±0.43</td>
<td>2.60±0.38</td>
<td>5.01±0.19 [100]</td>
<td>1.79±0.17</td>
</tr>
<tr>
<td>C. Cycle Power Plant</td>
<td>5.51±0.09</td>
<td>4.13±0.09</td>
<td>8.00±0.19</td>
<td>3.98±0.13</td>
<td>4.06±0.11 [50]</td>
<td>3.76±0.15</td>
</tr>
<tr>
<td>Bike Sharing (1)</td>
<td>36.45±0.46</td>
<td>32.67±0.81</td>
<td>34.93±0.97</td>
<td>18.89±0.36</td>
<td>14.81±0.44 [100]</td>
<td>15.17±0.44</td>
</tr>
<tr>
<td>Bike Sharing (2)</td>
<td>122.65±2.87</td>
<td>113.18±1.73</td>
<td>117.25±2.01</td>
<td>42.06±2.06</td>
<td>38.69±1.24 [100]</td>
<td>36.93±1.19</td>
</tr>
<tr>
<td>Phys. Prop.</td>
<td>5.19±0.03</td>
<td>4.91±1.26</td>
<td>6.49±1.15</td>
<td>4.40±0.04</td>
<td>4.20±0.05 [100]</td>
<td>4.21±0.04</td>
</tr>
</tbody>
</table>

Results: Missing data

- Randomly hide 30% of the data
- Mean and mode imputation for baselines

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RR</th>
<th>SVR (RBF)</th>
<th>SVR (polynomial)</th>
<th>DT</th>
<th>MLP (5 Layer)</th>
<th>CSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy Eff. (1)</td>
<td>3.01±0.15</td>
<td>3.38±0.27</td>
<td>6.88±0.63</td>
<td>2.57±0.49</td>
<td>2.49±0.48 [10]</td>
<td>2.17±0.25</td>
</tr>
<tr>
<td>Energy Eff. (2)</td>
<td>3.26±0.16</td>
<td>3.57±0.30</td>
<td>6.65±0.48</td>
<td>2.64±0.28</td>
<td>3.02±0.36 [10]</td>
<td>2.48±0.22</td>
</tr>
<tr>
<td>C. Comp. Strength</td>
<td>10.33±0.61</td>
<td>11.39±0.48</td>
<td>13.16±1.17</td>
<td>9.90±1.05</td>
<td>10.01±0.54 [10]</td>
<td>9.69±0.79</td>
</tr>
<tr>
<td>SkillCraft Master Table</td>
<td>1.79±1.63</td>
<td>1.05±0.03</td>
<td>1.61±0.33</td>
<td>1.08±0.03</td>
<td>1.10±0.04 [10]</td>
<td>1.05±0.01</td>
</tr>
<tr>
<td>Abalone</td>
<td>2.27±0.07</td>
<td>2.31±0.08</td>
<td>3.12±0.79</td>
<td>2.42±0.07</td>
<td>2.28±0.07 [10]</td>
<td>2.40±0.13</td>
</tr>
<tr>
<td>Wine Quality</td>
<td>0.76±0.02</td>
<td><strong>0.73±0.02</strong></td>
<td>0.93±0.21</td>
<td>0.78±0.02</td>
<td>0.76±0.03 [10]</td>
<td>0.78±0.02</td>
</tr>
<tr>
<td>Parkinsons Tel. (1)</td>
<td>7.52±0.11</td>
<td>6.91±0.13</td>
<td>8.12±0.11</td>
<td><strong>3.10±0.22</strong></td>
<td>5.90±0.28 [10]</td>
<td><strong>4.98±0.12</strong></td>
</tr>
<tr>
<td>Parkinsons Tel. (2)</td>
<td>9.76±0.18</td>
<td>9.38±0.21</td>
<td>10.68±0.23</td>
<td><strong>3.59±0.81</strong></td>
<td>7.67±0.18 [10]</td>
<td><strong>6.58±0.18</strong></td>
</tr>
<tr>
<td>C. Cycle Power Plant</td>
<td>5.51±0.09</td>
<td>6.16±0.15</td>
<td>10.45±0.31</td>
<td><strong>5.29±0.36</strong></td>
<td>5.33±0.07 [50]</td>
<td><strong>5.04±0.12</strong></td>
</tr>
<tr>
<td>Bike Sharing (1)</td>
<td>37.40±0.52</td>
<td>35.50±0.31</td>
<td>36.85±0.38</td>
<td>25.41±1.5</td>
<td><strong>21.51±0.83 [50]</strong></td>
<td><strong>23.89±0.19</strong></td>
</tr>
<tr>
<td>Bike Sharing (2)</td>
<td>123.81±1.26</td>
<td>127.06±1.55</td>
<td>130.20±1.13</td>
<td><strong>71.93±1.18</strong></td>
<td><strong>64.03±1.66 [50]</strong></td>
<td><strong>75.65±1.51</strong></td>
</tr>
<tr>
<td>Phys. Prop.</td>
<td>5.18±0.02</td>
<td>7.53±0.67</td>
<td>7.87±0.83</td>
<td>5.08±0.03</td>
<td><strong>4.99±0.09 [100]</strong></td>
<td><strong>4.70±0.03</strong></td>
</tr>
</tbody>
</table>

## Grade prediction

- **Baselines:** Grade Point Average (GPA), Biased Matrix Factorization

<table>
<thead>
<tr>
<th>Dataset</th>
<th>GPA</th>
<th>BMF</th>
<th>CSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI-1</td>
<td>0.52±0.02</td>
<td>0.48±0.03</td>
<td>0.48±0.03</td>
</tr>
<tr>
<td>CSCI-2</td>
<td>0.56±0.02</td>
<td>0.55±0.02</td>
<td>0.55±0.03</td>
</tr>
<tr>
<td>CSCI-3</td>
<td>0.48±0.04</td>
<td>0.48±0.04</td>
<td>0.48±0.05</td>
</tr>
<tr>
<td>CSCI-4</td>
<td>0.53±0.03</td>
<td>0.52±0.04</td>
<td>0.51±0.03</td>
</tr>
<tr>
<td>CSCI-5</td>
<td>0.43±0.02</td>
<td>0.43±0.02</td>
<td>0.42±0.02</td>
</tr>
<tr>
<td>CSCI-6</td>
<td>0.63±0.03</td>
<td>0.58±0.03</td>
<td>0.57±0.03</td>
</tr>
<tr>
<td>CSCI-7</td>
<td>0.57±0.02</td>
<td>0.58±0.01</td>
<td>0.56±0.02</td>
</tr>
<tr>
<td>CSCI-8</td>
<td>0.52±0.02</td>
<td>0.49±0.03</td>
<td>0.47±0.02</td>
</tr>
<tr>
<td>CSCI-9</td>
<td>0.61±0.03</td>
<td>0.60±0.05</td>
<td>0.57±0.03</td>
</tr>
<tr>
<td>CSCI-10</td>
<td>0.58±0.04</td>
<td>0.56±0.04</td>
<td>0.56±0.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>GPA</th>
<th>BMF</th>
<th>CSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI-11</td>
<td>0.68±0.06</td>
<td>0.66±0.04</td>
<td>0.67±0.03</td>
</tr>
<tr>
<td>CSCI-12</td>
<td>0.58±0.04</td>
<td>0.51±0.04</td>
<td>0.48±0.01</td>
</tr>
<tr>
<td>CSCI-13</td>
<td>0.67±0.03</td>
<td>0.55±0.05</td>
<td>0.54±0.03</td>
</tr>
<tr>
<td>CSCI-14</td>
<td>0.70±0.06</td>
<td>0.62±0.03</td>
<td>0.65±0.07</td>
</tr>
<tr>
<td>CSCI-15</td>
<td>0.56±0.03</td>
<td>0.56±0.06</td>
<td>0.57±0.03</td>
</tr>
<tr>
<td>CSCI-16</td>
<td>0.52±0.03</td>
<td>0.51±0.03</td>
<td>0.50±0.02</td>
</tr>
<tr>
<td>CSCI-17</td>
<td>0.60±0.02</td>
<td>0.58±0.05</td>
<td>0.59±0.05</td>
</tr>
<tr>
<td>CSCI-18</td>
<td>0.57±0.03</td>
<td>0.56±0.05</td>
<td>0.55±0.04</td>
</tr>
<tr>
<td>CSCI-19</td>
<td>0.68±0.04</td>
<td>0.70±0.04</td>
<td>0.61±0.04</td>
</tr>
<tr>
<td>CSCI-20</td>
<td>0.61±0.06</td>
<td>0.58±0.02</td>
<td>0.63±0.04</td>
</tr>
</tbody>
</table>

---

Recap
- Nonlinear system identification is tensor completion
- Low-rank models can model highly nonlinear functions
- Even if not low-rank: Identification of principal components

What about continuous inputs?
- One way to address the problem: quantization
- Coarse quantization: low resolution, poor generalization
- Fine quantization: high computational complexity, overfitting
- Alternative approach?
Fourier Series Representation

- Modeling non-periodic functions with compact support
  - Without loss of generality we may restrict the domain of $f$ on $[0,1]$
  - Using the even periodic extension of $f$ we have
    \[ f(x) = \alpha_0 + \sum_{k=1}^{\infty} \alpha_k \sqrt{2} \cos(k\pi x), \]
    using orthogonal basis functions $\phi_0(x) = 1$ and $\phi_k(x) = \sqrt{2} \cos(k\pi x)$, $k > 0$.

- Every periodic function which has finite energy over a single period has a unique Fourier representation that converges in $L^2$ to $f$ as $K \to \infty$

Fourier Series Representation

- Similarly for the multivariate case

\[ f(x) = \sum_{k_1=0}^{\infty} \cdots \sum_{k_N=0}^{\infty} \alpha_k \prod_{n=1}^{N} \phi_{k_n}(x[n]), \text{ where } k = (k_1, \ldots, k_N) \in \mathbb{N}^N. \]

- **Series estimator**

  - Truncated series with cutoffs \( K_1, \ldots, K_N \)

  \[ f(x) = \sum_{k_1=0}^{K_1-1} \cdots \sum_{k_N=0}^{K_N-1} \alpha_k \prod_{n=1}^{N} \phi_{k_n}(x[n]), \quad \alpha_k = \int_0^{1} \cdots \int_0^{1} f(x) \phi_k(x) dx, \quad (1) \]

  where \( \phi_k(x) = \prod_{n=1}^{N} \phi_{k_n}(x[n]). \)

  - Assuming that input samples \( \{x_m\} \) are drawn independently, a natural estimator for the coefficients is

  \[ \hat{x}[k_1, \ldots, k_N] = \hat{\alpha}_k = \frac{1}{M} \sum_{m=1}^{M} y_m \phi_k(x_m). \]

Generalized Canonical Polyadic Decomposition

- **Challenges**
  - Number of Fourier coefficient grows exponentially
  - Suitable only for small $N$
  - Large variance of the estimates when $M$ is small

- **Breaking the curse of dimensionality**
  - Low rank model of the coefficient tensor

\[
\mathcal{X}[k_1, \ldots, k_N] = \hat{\alpha}_k = \frac{1}{M} \sum_{m=1}^{M} y_m \phi_k(x_m), \quad \mathcal{X}[k_1, \ldots, k_N] = \sum_{r=1}^{R} \prod_{n=1}^{N} a^r_n[k_n].
\]

- Substituting back in Equation (1) we have that $f(x) = \sum_{r=1}^{R} f_{1,r}(x[1]) \cdots f_{N,r}(x[N])$.

- Each univariate function has a compact Fourier representation

Generalized Canonical Polyadic Decomposition

- Generalization of the CPD from tensors to functions
  - \(f\) is compactly supported in the domain \([0, 1]^N\).
  - Differentiable up to a certain order.
  - Low rank Fourier series coefficient tensor.
    - Controls complexity
    - Number of parameters \(O(K^N) \rightarrow O(KNR)\)

- Training the model:
  - Direct Optimization
    - Compute the coefficient tensor using sample averages.
    - We refer to this approach as (W)FSA-LR.
  - Hidden Tensor Factorization
    - Fit the coefficients directly on the training data.
    - We refer to this approach as FSA-HTF.

Results (synthetic data)

- **Exp. 1**
  - True function is low-rank (R=3)
  - Each univariate function is a linear combination of two Gaussians

- **Exp. 2**
  - True function is a neural network with a single hidden layer
  - Number of variables: 10
  - Generate 1000 samples, vary SNR
  - Set SNR=15, vary number of samples

Results (real data)

Linear Regression (LR), Polynomial Regression (PR), Support Vector Regression (SVR), Neural Network (NNET), Decision Tree (DT), low-rank RKHS model (AMP), Canonical System Identification (CSID).

MSE performance of standard baseline models.

<table>
<thead>
<tr>
<th>DATASET</th>
<th>LR</th>
<th>PR</th>
<th>SVR (RBF)</th>
<th>NNET</th>
<th>DT</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSAR</td>
<td>1.41 ± 0.31</td>
<td>1.41 ± 0.44</td>
<td><strong>1.19 ± 0.30</strong></td>
<td>1.42 ± 0.55</td>
<td>1.64 ± 0.37</td>
</tr>
<tr>
<td>EE [1]</td>
<td>8.15 ± 1.13</td>
<td>0.76 ± 0.08</td>
<td>4.69 ± 0.68</td>
<td>0.31 ± 0.08</td>
<td>0.30 ± 0.03</td>
</tr>
<tr>
<td>EE [2]</td>
<td>10.39 ± 1.84</td>
<td>2.87 ± 0.36</td>
<td>7.36 ± 1.66</td>
<td>2.03 ± 0.71</td>
<td>3.21 ± 0.26</td>
</tr>
<tr>
<td>ASN</td>
<td>23.27 ± 1.74</td>
<td>15.95 ± 1.21</td>
<td>11.09 ± 1.60</td>
<td>6.61 ± 1.24</td>
<td>6.45 ± 0.35</td>
</tr>
<tr>
<td>SMT</td>
<td>1.03 ± 0.06</td>
<td>1.02 ± 0.09</td>
<td>1.01 ± 0.05</td>
<td>1.03 ± 0.06</td>
<td>1.06 ± 0.04</td>
</tr>
<tr>
<td>AB</td>
<td>4.91 ± 0.30</td>
<td>4.54 ± 0.12</td>
<td>4.44 ± 0.21</td>
<td><strong>4.33 ± 0.22</strong></td>
<td>5.29 ± 0.22</td>
</tr>
<tr>
<td>CPP</td>
<td>19.57 ± 0.50</td>
<td>17.18 ± 0.41</td>
<td>15.60 ± 0.51</td>
<td>16.08 ± 0.50</td>
<td><strong>14.85 ± 0.97</strong></td>
</tr>
<tr>
<td>SC</td>
<td>315.93 ± 6.76</td>
<td>171.52 ± 5.41</td>
<td>213.60 ± 6.71</td>
<td><strong>125.12 ± 8.13</strong></td>
<td>130.56 ± 8.35</td>
</tr>
<tr>
<td>PP</td>
<td>27.10 ± 0.27</td>
<td>25.05 ± 1.06</td>
<td>21.24 ± 0.42</td>
<td>17.55 ± 0.38</td>
<td>19.24 ± 0.23</td>
</tr>
</tbody>
</table>

MSE performance of CPD-based models.

<table>
<thead>
<tr>
<th>DATASET</th>
<th>AMP</th>
<th>CSID</th>
<th>FSA-HTF</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSAR</td>
<td>1.65 ± 0.50</td>
<td>1.48 ± 0.17</td>
<td>1.38 ± 0.53</td>
</tr>
<tr>
<td>EE [1]</td>
<td>0.31 ± 0.06</td>
<td>0.18 ± 0.02</td>
<td>0.16 ± 0.10</td>
</tr>
<tr>
<td>EE [2]</td>
<td>0.46 ± 0.10</td>
<td>0.34 ± 0.06</td>
<td><strong>0.24 ± 0.09</strong></td>
</tr>
<tr>
<td>ASN</td>
<td>8.73 ± 1.29</td>
<td><strong>3.05 ± 0.53</strong></td>
<td>4.01 ± 0.32</td>
</tr>
<tr>
<td>SMT</td>
<td>0.99 ± 0.05</td>
<td>0.98 ± 0.06</td>
<td><strong>0.94 ± 0.06</strong></td>
</tr>
<tr>
<td>AB</td>
<td>4.75 ± 0.3</td>
<td>4.95 ± 0.21</td>
<td>4.52 ± 0.30</td>
</tr>
<tr>
<td>CPP</td>
<td>N/A</td>
<td>15.25 ± 0.45</td>
<td>15.00 ± 0.66</td>
</tr>
<tr>
<td>SC</td>
<td>N/A</td>
<td>N/A</td>
<td>127.76 ± 11.86</td>
</tr>
<tr>
<td>PP</td>
<td>N/A</td>
<td>18.21 ± 0.45</td>
<td><strong>16.77 ± 0.35</strong></td>
</tr>
</tbody>
</table>

Take-home points

- Nonlinear system identification is tensor completion.
- Provably correct system identification is possible under low rank conditions.
- Low-rank models can model highly nonlinear functions.
- Even if not low-rank: Identification of principal components of nonlinear mapping.

- Extension of CPD from tensors to multivariate functions.
  - Compactly supported functions.
  - Differentiable up to certain order.
References


• Rendle, S. 2010. “Factorization machines”. In IEEE International Conference on Data Mining, 995–1000.


• Huang, B., Mu, C., Goldfarb, D., and Wright, J. 2014. “Provable low-rank tensor recovery”.
