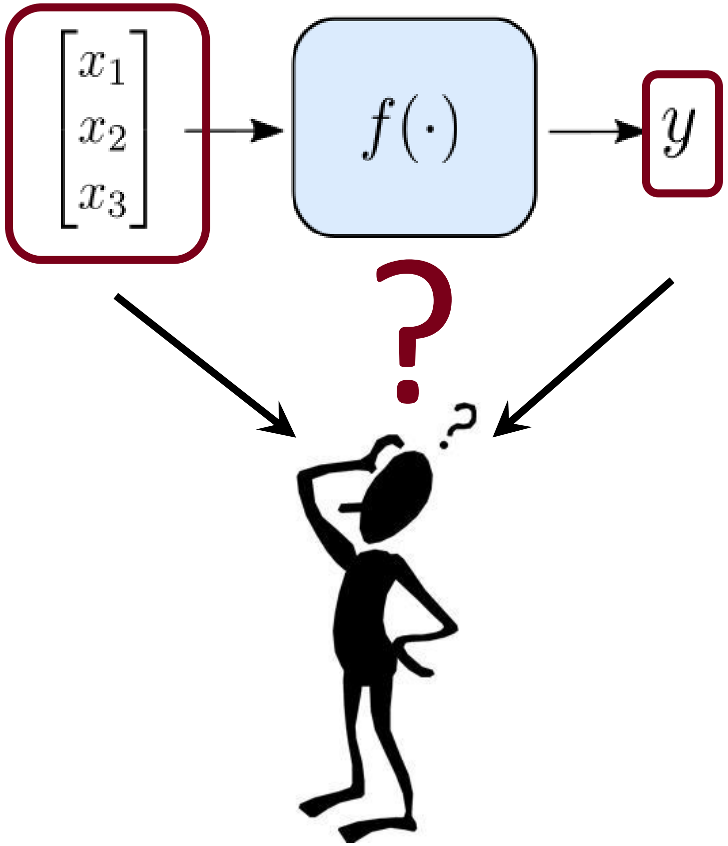
Nonlinear System Identification via Tensor Completion

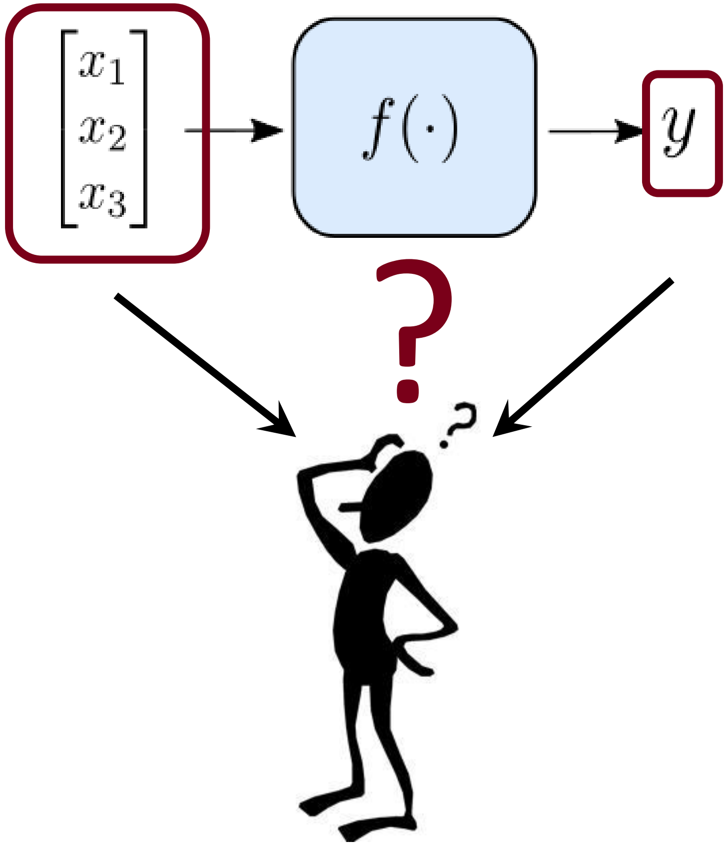
N. Kargas and N. D. Sidiropoulos

The Supervised Learning Problem



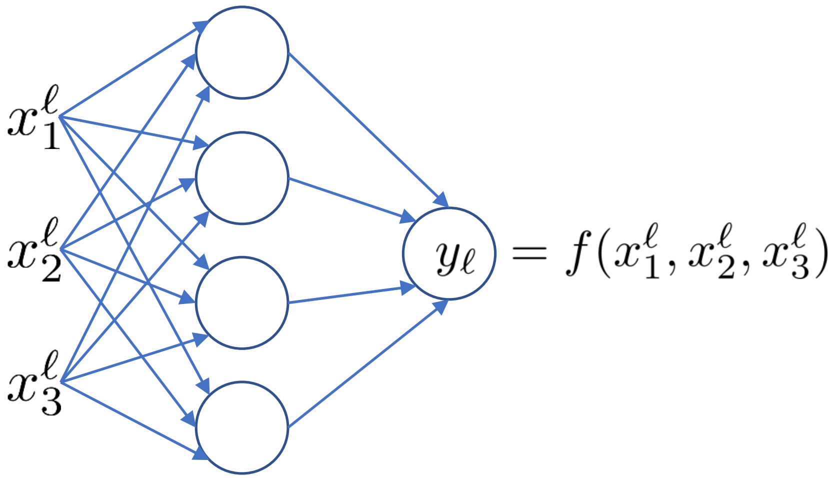
Categorical (classification, binary or FA)
Real-valued (prediction, regression)
Complex-valued (channel; MRI k-space)

AKA: I/O (Nonlinear) System Identification



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

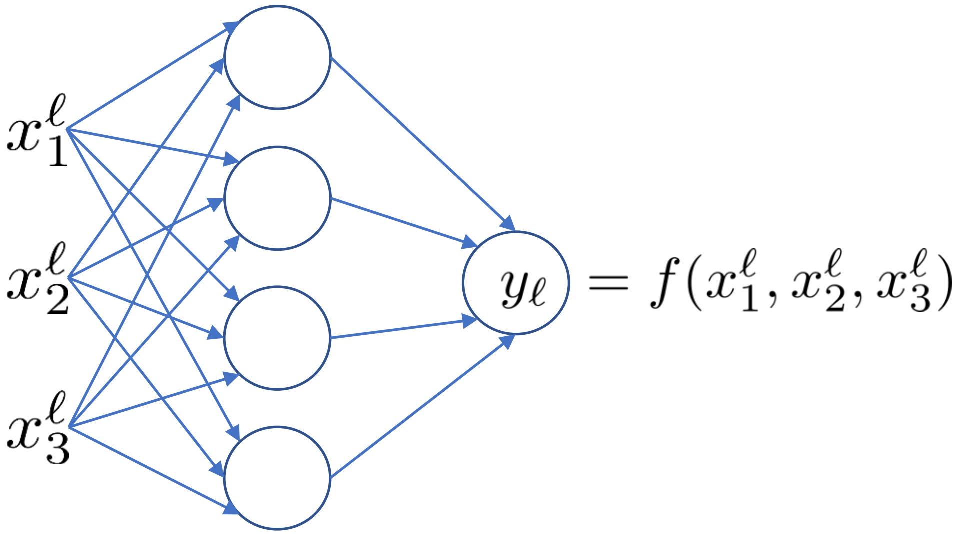
(Deep) Neural Networks



- 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

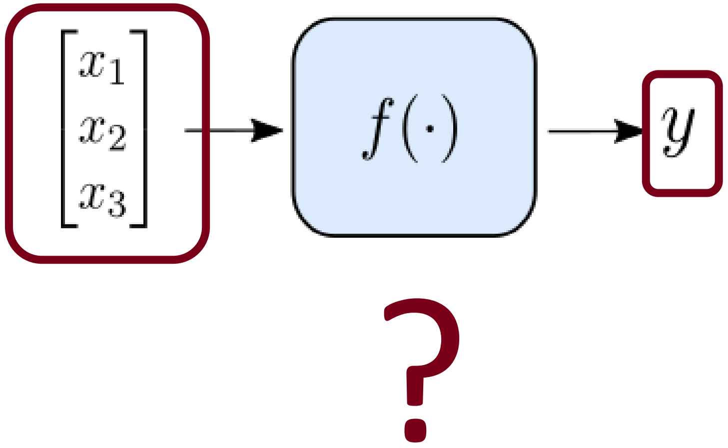
- Most popular method for learning to mimic nonlinear functions
- Some theory … but, for most part …
  - Don't understand why they work so well
  - Choosing architecture is art
  - Hard to interpret
- Against all odds and principles!



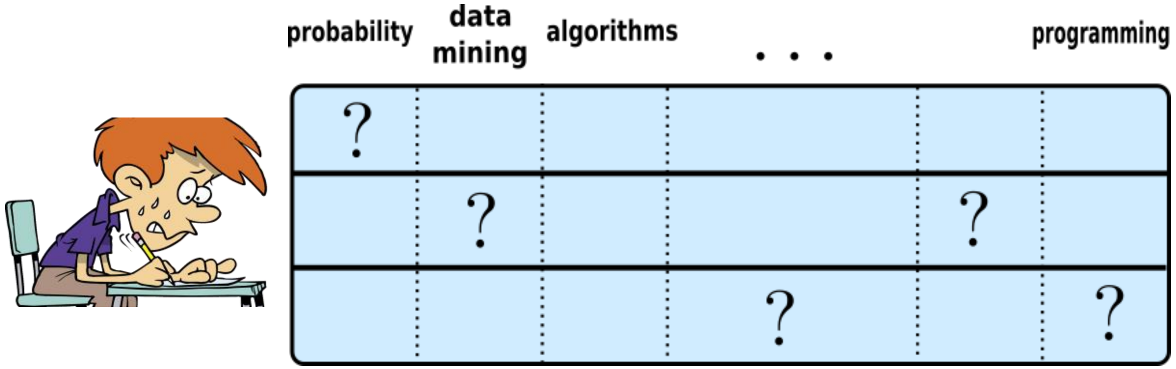
- This talk: principled alternative
- Based on tensor principal components
- Advantages: `universal', intuitive, interpretable, backed by theory
- Works with incomplete input data – important in practice

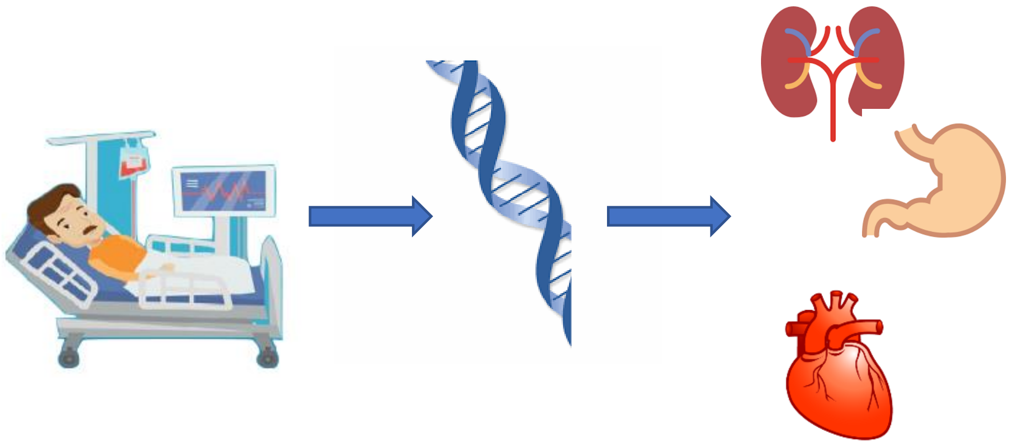
Introduction

- 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

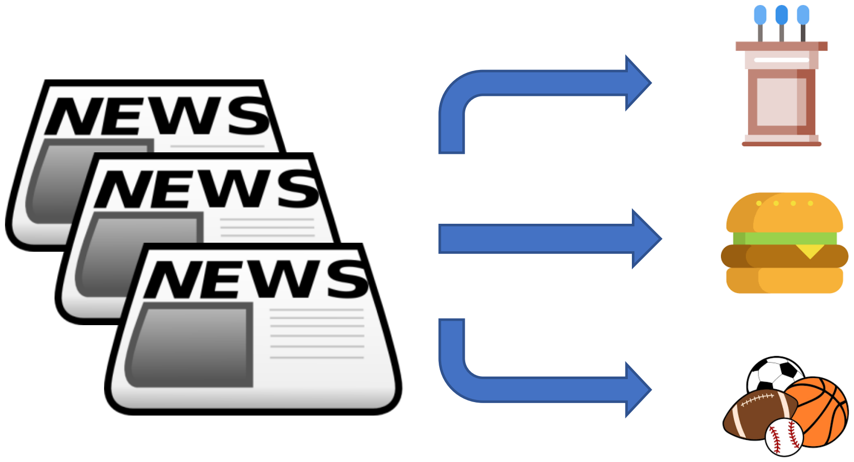




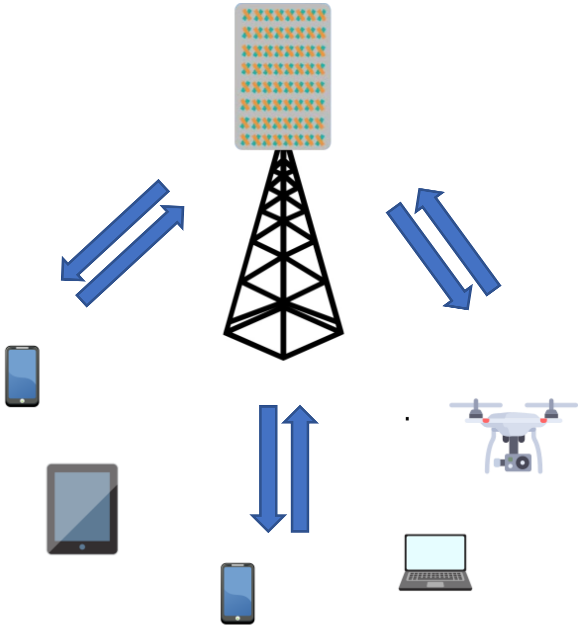
Course grade prediction

Drug response prediction

Motivation



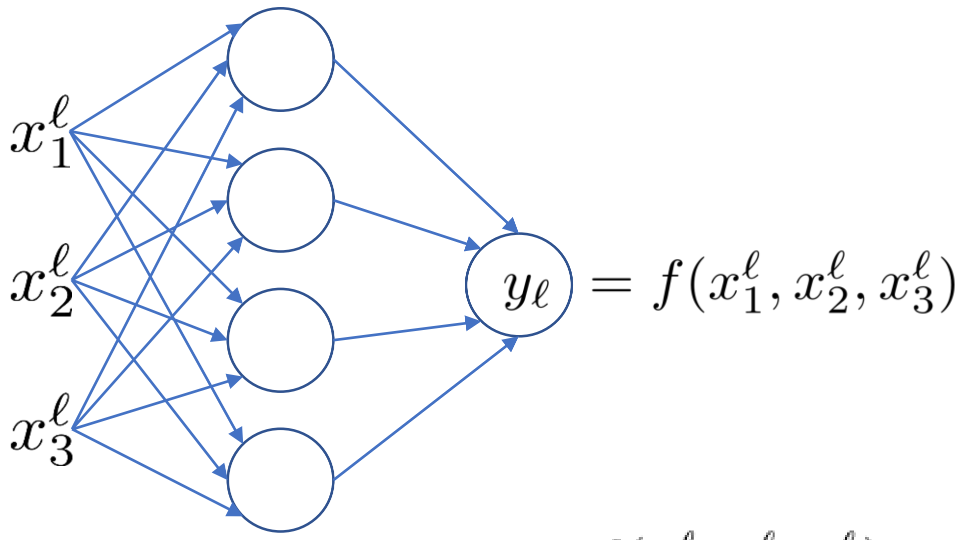
Text classification

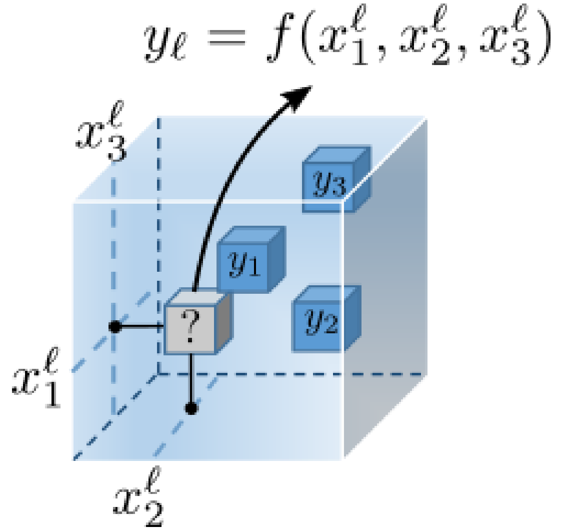


Channel estimation

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



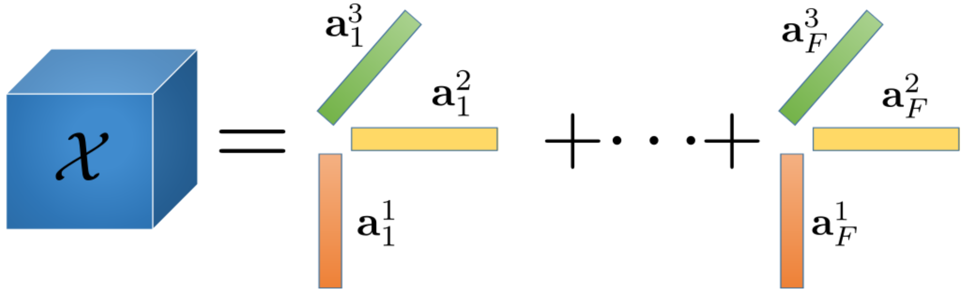


Canonical Polyadic Decomposition (CPD)

- 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

$$\mathcal{X} = \sum_{f=1}^{F} \mathbf{a}_f^1 \circ \mathbf{a}_f^2 \circ \cdots \circ \mathbf{a}_f^N$$

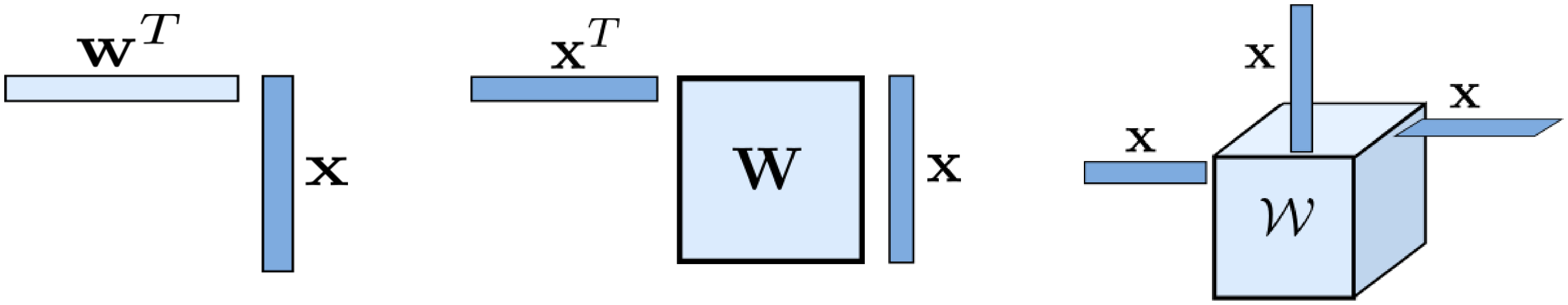
- Tensor rank is smallest *F* for which such decomposition exists → *Canonical*



- Element-wise: $\mathcal{X}(i_1, \ldots, i_N) = \sum_{f=1}^{F} \prod_{n=1}^{N} \mathbf{a}_f^n(i_n)$
- 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: $\mathrm{vec}(\mathcal{X}) = (\mathbf{A}_N \odot \cdots \odot \mathbf{A}_1)\mathbf{1}$

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



$$y = \mathbf{w}^T\mathbf{x} \qquad y = \mathbf{x}^T\mathbf{W}\mathbf{x} \qquad y = \mathcal{W} \times_1 \mathbf{x} \times_2 \mathbf{x} \times_3 \mathbf{x}$$

Prior work

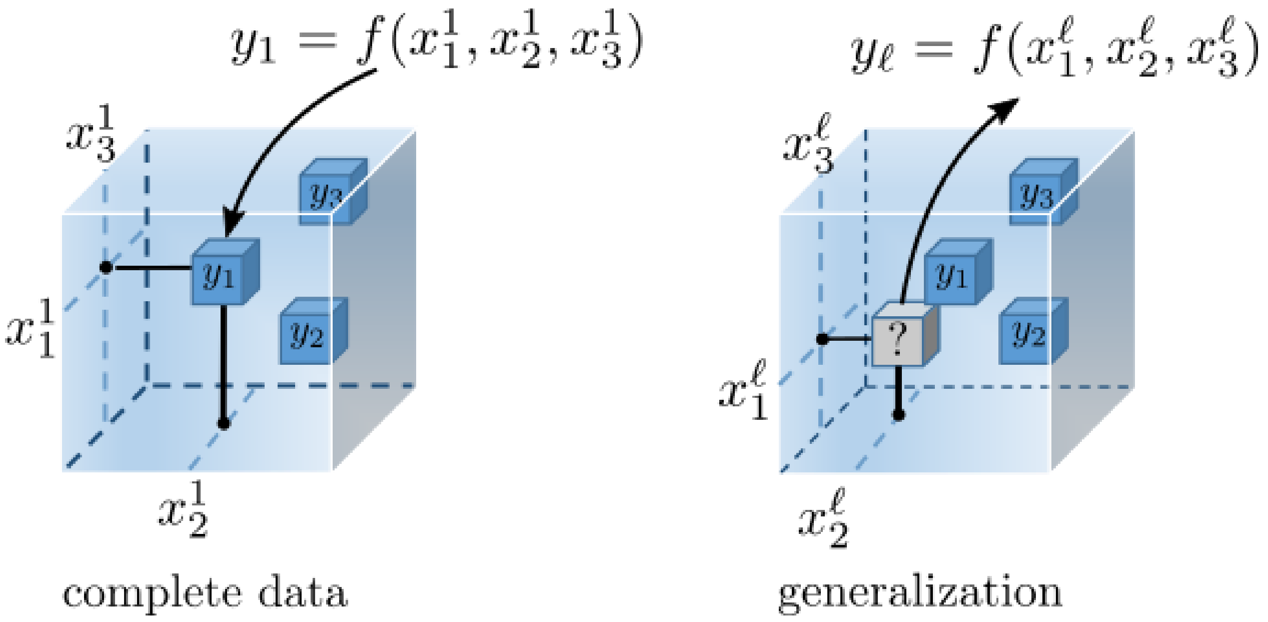
- Number of parameters grows exponentially with the order d

  → 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

Canonical System Identification (CSID)

- <u>We propose:</u>
  - <u>Single high-order tensor</u> for learning a general nonlinear system



Canonical System Identification (CSID)

- Claims:

  - CPD can model *any* nonlinearity (even of ∞ 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 $\Longrightarrow$ 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., $f(x_1, \ldots, x_N) = \prod_{n=1}^{N} \mathrm{sign}(x_n)$
- Sum of separable functions: $y = f(x_1, \ldots, x_N) = \sum_{n=1}^{N} f_n(x_n)$
  - Maximal rank: N

    e.g., $f(x_1, \ldots, x_N) = \sum_{n=1}^{N} \mathrm{sign}(x_n)$
- 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 $[\mathbf{x}_m(1), \ldots, \mathbf{x}_m(N)]$ is viewed as a cell multi-index and the cell content is the estimated response of the system:

$$\min_{\mathcal{X}} \frac{1}{M} \sum_{m=1}^{M} \left(y_m - \mathcal{X}\left(\mathbf{x}_m(1), \ldots, \mathbf{x}_m(N)\right)\right)^2$$

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

$$\begin{aligned} \min_{\mathcal{X}, \{\mathbf{A}_n\}_{n=1}^{N}} \quad & \frac{1}{M} \sum_{m=1}^{M} \left(y_m - \mathcal{X}(\mathbf{x}_m(1), \ldots, \mathbf{x}_m(N))\right)^2 + \sum_{n=1}^{N} \rho \|\mathbf{A}_n\|_F^2 \\ \text{subject to} \quad & \mathcal{X} = \sum_{f=1}^{F} \mathbf{A}_1(:, f) \odot \cdots \odot \mathbf{A}_N(:, f) \end{aligned}$$

Handling ordinal features

- Datasets often contain both categorical and ordinal predictors.

$$\min_{\mathcal{X}, \{\mathbf{A}_n\}_{n=1}^N} \frac{1}{M} \left\| \sqrt{\mathcal{W}} \circledast (\mathcal{Y} - \mathcal{X}) \right\|_F^2 + \sum_{n=1}^N \rho \|\mathbf{A}_n\|_F^2 + \sum_{n=1}^N \mu_n \|\mathbf{T}_n \mathbf{A}_n\|_F^2$$

$$\text{subject to} \quad \mathcal{X} = \sum_{f=1}^F \mathbf{A}_1(:, f) \odot \cdots \odot \mathbf{A}_N(:, f),$$

where

$$\mathbf{T}_n = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{bmatrix} \quad \text{or} \quad \mathbf{T}_n = \begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \end{bmatrix}$$

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
  - M. Bousse, N. Vervliet, I. Domanov, O. Debals, and L. De Lathauwer, "Linear systems with a canonical polyadic decomposition constrained solution: Algorithms and applications", *Numerical Linear Algebra with Applications*, vol. 25, no. 6, Aug. 2018.
- … 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 $O(\sqrt{FI^N}\log(N))$, using
  - M. Yuan C. Zhang, "On Tensor Completion via Nuclear Norm Minimization", *Foundations Computational Mathematics*, vol. 16, no. 4, Aug. 2016.

Algorithm

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

$$\min_{\mathcal{X},\{\mathbf{A}_n\}_{n=1}^N} \frac{1}{M}\left\|\sqrt{\mathcal{W}} \circledast (\mathcal{Y}-\mathcal{X})\right\|_F^2 + \sum_{n=1}^N \rho\|\mathbf{A}_n\|_F^2 + \sum_{n=1}^N \mu_n \|\mathbf{T}_n\mathbf{A}_n\|_F^2$$

$$\text{subject to} \quad \mathcal{X} = \sum_{f=1}^F \mathbf{A}_1(:,f)\odot\cdots\odot\mathbf{A}_N(:,f),$$

- Large scale problems ⟹ SGD, Block-stochastic GD

Missing data

- Let $\mathcal{O}$ and $\mathcal{M}$ denote the indices of the observed and missing entries of a single observation

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

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

$$\begin{aligned} f(\mathbf{x}_{\mathcal{O}}) &= \mathbb{E}_{\mathbf{x}_{\mathcal{M}}|\mathbf{x}_{\mathcal{O}}}[f(\mathbf{x}_{\mathcal{O}}, \mathbf{x}_{\mathcal{M}})] = \mathcal{X}(i_1, \ldots, i_T, :, \ldots, :) \times_{T+1} \mathbf{p}_{T+1} \cdots \times_{T+L} \mathbf{p}_N \\ &= \sum_{f=1}^{F} \prod_{n=1}^{T} \mathbf{A}_n(i_n, f) \prod_{n=T+1}^{N} \mathbf{p}_n^T \mathbf{A}_n(:, f) \end{aligned}$$

Multi-output regression

- No correlation between the K output variables $\Rightarrow$ build K independent models
- Output variables are usually correlated

- <u>Better approach:</u>
  - 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: $\mathcal{X}(i_1, \ldots, i_N, :) = (\mathbf{A}_1(i_1, :) \circledast \cdots \circledast \mathbf{A}_N(i_N, :))\, \mathbf{V}^T$

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

Dataset	N	M	Type	Range
Concrete Compressive Strength	8	1030	Ordinal	$y \in (2, 83)$
SkillCraft Master Table	18	3337	Ordinal	$y \in (1, 7)$
Abalone	8	4177	Mixed	$y \in (1, 29)$
Wine Quality	11	4898	Ordinal	$y \in (3, 9)$
Combined Cycle Power Plant	4	9568	Ordinal	$y \in (420, 496)$
Physicochemical Properties	9	45730	Ordinal	$y \in (0, 21)$
Energy efficiency (2)	8	788	Ordinal	$y_1 \in (6, 44)$ $y_2 \in (10, 49)$
Parkinsons Telemonitoring (2)	19	5875	Mixed	$y_1 \in (5, 40)$ $y_2 \in (7, 55)$
Bike Sharing (2)	12	17379	Mixed	$y_1 \in (0, 367)$ $y_2 \in (0, 886)$

Dataset	N	M	Sparsity
CSCI-1	34	996	0.54
CSCI-2	34	990	0.55
CSCI-3	34	983	0.55
CSCI-4	34	958	0.55
CSCI-5	34	953	0.56
CSCI-6	34	931	0.56
CSCI-7	34	911	0.56
CSCI-8	34	898	0.56
CSCI-9	34	867	0.56
CSCI-10	34	856	0.57

Dataset	N	M	Sparsity
CSCI-11	34	704	0.57
CSCI-12	34	696	0.58
CSCI-13	34	650	0.57
CSCI-14	34	636	0.59
CSCI-15	34	600	0.57
CSCI-16	34	598	0.57
CSCI-17	34	529	0.56
CSCI-18	34	519	0.55
CSCI-19	34	431	0.55
CSCI-20	34	403	0.55

Results: Full data

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

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

Results: Missing data

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

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

Results: Multiple outputs

- 2 output variables for each dataset

Dataset	RR	MLP (1 Layer)	MLP (3 Layer)	MLP (5 Layer)	DT	CSID
En. Eff. (2)	2.70±0.19	2.82±0.08 [50]	2.73±0.11[100]	2.67±0.11[10]	**2.19±0.19**	**2.01±0.14**
Park. Tel. (2)	12.19±0.09	7.59±0.21[250]	6.54±0.06[250]	6.18±0.42[250]	**3.37±0.39**	**2.85±0.22**
B. Shar. (2)	127.75±3.32	64.12±6.49[250]	43.60±1.95[100]	**42.25±1.22[100]**	46.21±1.20	**45.29±1.47**

Grade prediction

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

Dataset	GPA	BMF	CSID
CSCI-1	0.52±0.02	**0.48±0.03**	**0.48±0.03**
CSCI-2	0.56±0.02	**0.55±0.02**	**0.55±0.03**
CSCI-3	**0.48±0.04**	**0.48±0.04**	**0.48±0.05**
CSCI-4	0.53±0.03	0.52±0.04	**0.51±0.03**
CSCI-5	0.43±0.02	0.43±0.02	**0.42±0.02**
CSCI-6	0.63±0.03	0.58±0.03	**0.57±0.03**
CSCI-7	0.57±0.02	0.58±0.01	**0.56±0.02**
CSCI-8	0.52±0.02	0.49±0.03	**0.47±0.02**
CSCI-9	0.61±0.03	0.60±0.05	**0.57±0.03**
CSCI-10	0.58±0.04	**0.56±0.04**	**0.56±0.04**

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

Take-home points

- ## Concluding remarks
  - 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 the nonlinear mapping

THANK YOU!

Questions?

References

- Kargas, N., and Sidiropoulos, N. D. “Nonlinear System Identification via Tensor Completion” (submitted) – see https://arxiv.org/pdf/1906.05746.pdf
- Rendle, S. 2010. “Factorization machines”. In IEEE International Conference on Data Mining, 995–1000.
- Huang, K., and Sidiropoulos, N. D. 2017. “Kullback-Leibler principal component for tensors is not NP-hard”. In Asilomar Conference on Signals, Systems, and Computers, 693–697.
- Krishnamurthy, A., and Singh, A. 2013. “Low-rank matrix and tensor completion via adaptive sampling”. In Advances in Neural Information Processing Systems, 836–844.
- Jain, P., and Oh, S. 2014. “Provable tensor factorization with missing data”. In Advances in Neural Information Processing Systems 27, 1431–1439.
- Huang, B., Mu, C., Goldfarb, D., and Wright, J. 2014. “Provable low-rank tensor recovery”.
- Sorensen, M., and De Lathauwer, L. 2019. “Fiber sampling approach to canonical polyadic decomposition and application to tensor completion”. SIAM Journal on Matrix Analysis and Applications 40(3):888–917.
- Kanatsoulis, C. I., Fu, X.; Sidiropoulos, N. D., and Akcakaya, M.2019. “Tensor completion from regular sub-Nyquist samples”. arXiv preprint arXiv:1903.00435. (to appear, IEEE Trans. on Signal Processing)