Scalable Low-Rank Nonlinear Subspace Tracking

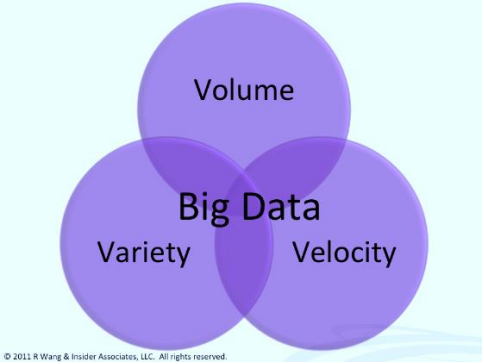
F. Sheikholeslami, D. Berberidis, and G. B. Giannakis

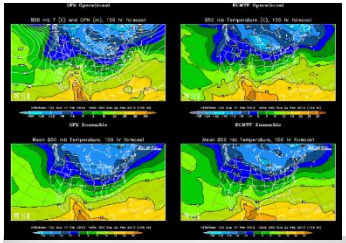
Dept. of ECE and DTC, University of Minnesota

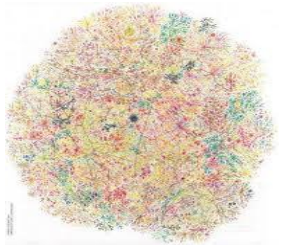
Acknowledgement: NSF grants 1500713, 1514056, and NIH grant 1R01GM104975-01

Learning from “Big Data”

- Challenges
  - Big size ($D \gg$ and/or $N \gg$)
  - Fast streaming
  - Incomplete
  - Noise and outliers
- Opportunities in key tasks
  - Dimensionality reduction
  - Online and robust regression, classification and clustering
  - Denoising and imputation







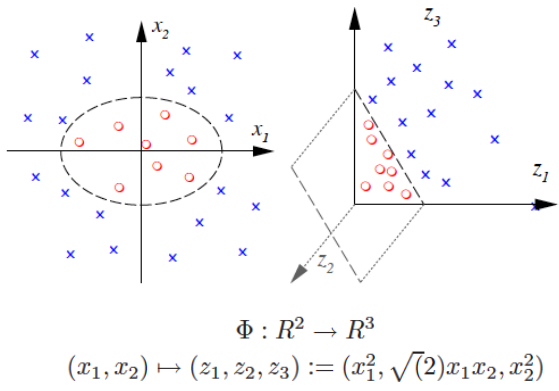
Outline

- Scalable kernel-based learning
  - Sparsity-aware low-rank approximation of lifted data
  - Online subspace tracking
  - Affordable memory and computation
  - Theoretical guarantees and test results
  - Future directions

Linear or nonlinear functions for learning?

- **Regression** or **classification**: Given $\{y_n, \mathbf{x}_n\}_{n=1}^N$, find $\hat{f}: \mathbf{x} \rightarrow y = f(\mathbf{x}) + v$
- Lift via nonlinear map $\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x}) \in \mathbb{R}^{\bar{D}}$ to linear $y_n = \boldsymbol{\phi}^\top(\mathbf{x}_n)\bar{\boldsymbol{\theta}} + v_n$
  - Pre-select kernel (inner product) function

$$\mathbf{x}_i^\top \mathbf{x}_j \rightarrow k(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\phi}(\mathbf{x}_j) \quad \text{e.g., } k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$$



  - RKHS basis expansion

$$\hat{f}(\mathbf{x}) = \sum_{n=1}^N \alpha_n k(\mathbf{x}, \mathbf{x}_n)$$

- Kernel-based nonparametric ridge regression

$$\mathbf{y} = \mathbf{K}\boldsymbol{\alpha} + \mathbf{v} \rightarrow \boldsymbol{\alpha} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1}\mathbf{y}$$

  - Memory requirement $\mathcal{O}(N^2)$, and complexity $\mathcal{O}(N^3)$

Prior art

❖ Functional gradient descent

- Tractable gradient descent (batch form in primal domain)
  - Pegasus [Shaleve-Schwart et al.'11]; Norma [Kivinen et al.'04]; [Wang et al. '12, Lu et al. '16]

❖ Kernel matrix approximation

- Tractable by leveraging low-rank attribute (in dual domain)
  - Nystrom approximation [Williamson-Seeger'00, Si et al. '14, Wang et al. '14]
  - Incomplete Cholesky decomposition [Fine-Scheinberg'01]

❖ Random feature approximation

- Kernel approximation via feature approximation
  - Random sampling and kernel matrix factorization [Rahimi-Recht'08, Zhang et al.'11]

❖ Reduced-dimensionality features

- Tractability through online extraction of low-dimensional features
  - Online (kernel) PCA [Honnein'012]; linear subspace tracking [Mardani et al.'15]

Low-rank approximation of mapped features

- Nonlinear mapping $\mathbf{x} \to \boldsymbol{\phi}(\mathbf{x})$
- Low-rank approximation $\boldsymbol{\phi}(\mathbf{x}) \simeq \hat{\boldsymbol{\phi}}(\mathbf{x}) := \bar{\mathbf{L}}\mathbf{q}$ $\bar{\mathbf{L}} \in \mathbb{R}^{\bar{D} \times r}$ $\mathbf{q} \in \mathbb{R}^{r \times 1}$

$$\min_{\bar{\mathbf{L}}, \{\mathbf{q}_i\}_{i=1}^N} \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{\phi}(\mathbf{x}_i) - \bar{\mathbf{L}}\mathbf{q}_i\|_{\mathcal{H}}^2 + \frac{\lambda}{N} \sum_{i=1}^{N} \|\mathbf{q}_i\|_2^2$$

- Alternating minimization

S1. Update virtual subspace $\bar{\mathbf{L}}$ while fixing feature vectors $\{\mathbf{q}_i\}_{i=1}^N$

$$\bar{\mathbf{L}} = \boldsymbol{\Phi}_N \mathbf{Q}_N^{\dagger} := \boldsymbol{\Phi}_N \mathbf{A} \qquad \boldsymbol{\Phi}_N := [\boldsymbol{\phi}(\mathbf{x}_1), \ldots, \boldsymbol{\phi}(\mathbf{x}_N)]$$
$$\mathbf{Q}_N := [\mathbf{q}_1, \ldots, \mathbf{q}_N]$$

S2. Update features $\{\mathbf{q}_i\}_{i=1}^N$ while fixing subspace $\bar{\mathbf{L}}$

$$\mathbf{q}_i = (\mathbf{A}^{\top}\mathbf{K}\mathbf{A} + \lambda \mathbf{I}_r)^{-1} \mathbf{A}^{\top} \mathbf{k}_i$$

Sparsity-aware low-rank approximation

- Row-sparsity of $\mathbf{A} \Rightarrow$ fewer "basis vectors" for $\bar{\mathbf{L}}$

$$\min_{\mathbf{A},\{\mathbf{q}_i\}_{i=1}^N} \frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{\phi}(\mathbf{x}_i)-\boldsymbol{\Phi}_N\mathbf{A}\mathbf{q}_i\|_2^2+\frac{\lambda}{N}\sum_{i=1}^{N}\|\mathbf{q}_i\|_2^2+\eta\sum_{i=1}^{N}\|\mathbf{a}_i\|_2$$

- BCD

S1. Update $\{\mathbf{q}_i\}_{i=1}^N$ with $\bar{\mathbf{L}}$ fixed $\qquad \mathbf{q}_i=(\mathbf{A}^\top\mathbf{K}\mathbf{A}+\lambda\mathbf{I}_r)^{-1}\mathbf{A}^\top\mathbf{k}_i$

S2. Update $\mathbf{A}$ with $\{\mathbf{q}_i\}_{i=1}^N$ fixed

$$\mathbf{a}_i[k+1]=\arg\min_{\mathbf{a}_i}\frac{1}{2}\mathbf{a}_i^\top\mathbf{H}_i\mathbf{a}_i+\mathbf{p}_i^\top\mathbf{a}_i+\eta\|\mathbf{a}_i\|_2$$

Group shrinkage

- Solvers
  - **i. BCD-Exact**: Exact minimization- more expensive, larger descent
  - **ii. BCD-PG**: Proximal gradient-based update, inexpensive, smaller descent

F. Sheikholeslami and G. B. Giannakis, "Scalable Kernel-based Learning via Low-rank Approximation of Lifted Data," *Proc. of 55th Allerton Conf. on Comm., Control, and Computing*, Oct. 4-6, 2017.

Convergence and generalization bounds

Proposition 1 Sequences $\{\mathbf{Q}[k], \mathbf{A}[k]\}_{k=1}^{\infty}$ generated by BCD-Ex and BCD-PG iterations converge to a stationary point.

❑ Generalization: performance guarantees on unseen data

- Equivalent optimization

$$\min_{\mathbf{A}\in\mathcal{A}} \frac{1}{N}\sum_{i=1}^{N} \underbrace{\min_{\mathbf{q}_i\in\mathcal{Q}} \|\boldsymbol{\phi}(\mathbf{x}_i) - \boldsymbol{\Phi}_N \mathbf{A}\mathbf{q}_i\|_{\mathcal{H}}^2}_{=:\ \ell(\mathbf{x},\bar{\mathbf{L}})}$$

$$\mathcal{Q} := \{\mathbf{q}\,|\,\mathbf{q}\in\mathbb{R}^r\ , \|\mathbf{q}\|_2 < B\}$$
$$\mathcal{A} := \{\mathbf{A}|\ \textstyle\sum_{i=1}^{N} \|\mathbf{a}_i\|_2 \le \tau\}$$

- Define $\|\mathcal{L}\|_{\mathcal{Q}} := \sup_{\bar{\mathbf{L}}\in\mathcal{L}} \|\bar{\mathbf{L}}\|_{\mathcal{Q}} = \sup_{\bar{\mathbf{L}}\in\mathcal{L}} \sup_{\mathbf{q}\in\mathcal{Q}} \|\bar{\mathbf{L}}\mathbf{q}\|_{\mathcal{H}}$

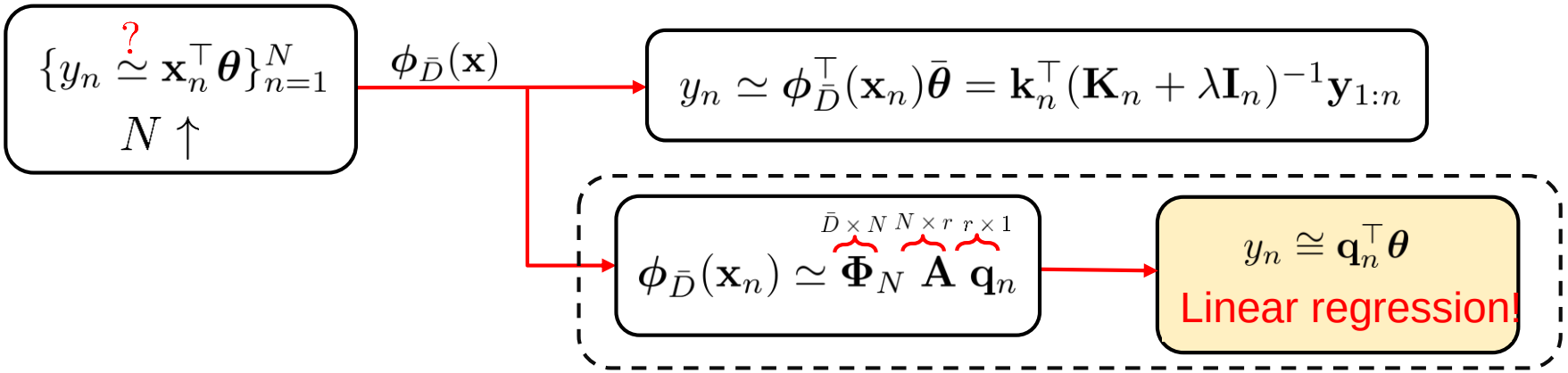
$$\mathcal{L} := \{\bar{\mathbf{L}}\,|\,\bar{\mathbf{L}} = \boldsymbol{\Phi}_N\mathbf{A}, \mathbf{A}\in\mathcal{A}\}$$

Proposition 2. If $\|\mathcal{L}\|_{\mathcal{Q}} \ge 1$, and $\kappa(\mathbf{x},\mathbf{x}) \le \kappa$, then for $\delta > 0$, it holds wp > $1-\delta$

$$\mathbb{E}[\ell(\mathbf{x};\bar{\mathbf{L}})] - 1/N\sum_{i=1}^{N}\ell(\mathbf{x}_i;\bar{\mathbf{L}}) \le \frac{d}{\sqrt{N}}\left(14\kappa B\tau + \frac{\kappa}{2}\sqrt{\ln(16N\kappa^2B^2\tau^2)}\right) + \kappa\sqrt{\frac{\ln(1/\delta)}{2N}} \quad \forall\bar{\mathbf{L}}\in\mathcal{L}$$

F. Sheikholeslami and G. B. Giannakis, "Scalable Kernel-based Learning via Low-rank Approximation of Lifted Data," *Proc. Allerton*, 2017.

Linearized kernel regression and classification



- Kernel matrix approximation

Proposition 3. If $e_i := \|\boldsymbol{\phi}(\mathbf{x}_i) - \hat{\boldsymbol{\phi}}(\mathbf{x}_i)\|_{\mathcal{H}}^2$ iid with $\bar{e} := \mathbb{E}[e_i]$ kernel matrix $\mathbf{K} = \boldsymbol{\Phi}^\top \boldsymbol{\Phi}$ can be approximated as $\hat{\mathbf{K}} = \hat{\boldsymbol{\Phi}}^\top \hat{\boldsymbol{\Phi}}$, and wp > $1 - 2e^{-2N\tau^2}$

$$\frac{1}{N}\|\mathbf{K} - \hat{\mathbf{K}}\|_F \leq \sqrt{\bar{e}+\tau}(\sqrt{\bar{e}+\tau}+2)$$

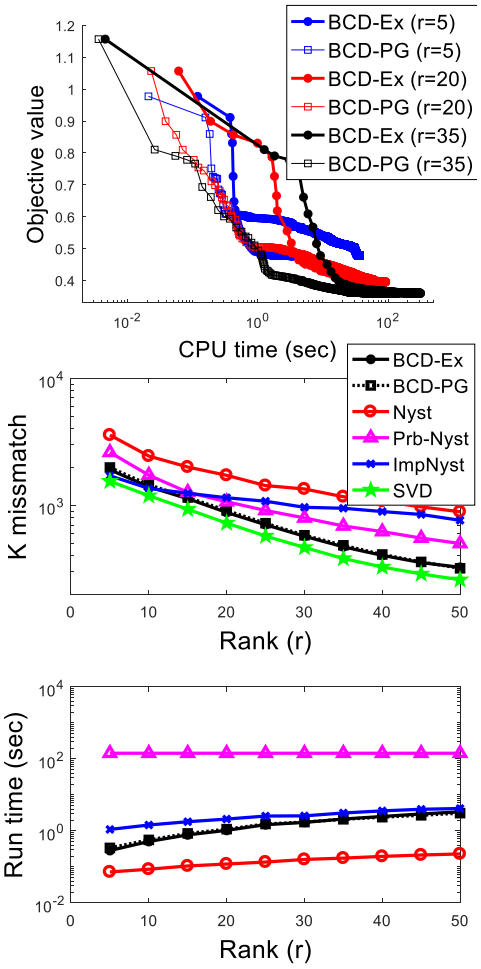
- Bounds also on support vector machines for regression and classification

F. Sheikholeslami, D. K. Berberidis, and G. B. Giannakis, "Kernel-based Low-rank Feature Extraction on a Budget for Big Data Streams," arxiv:1601.07947.

Simulation tests

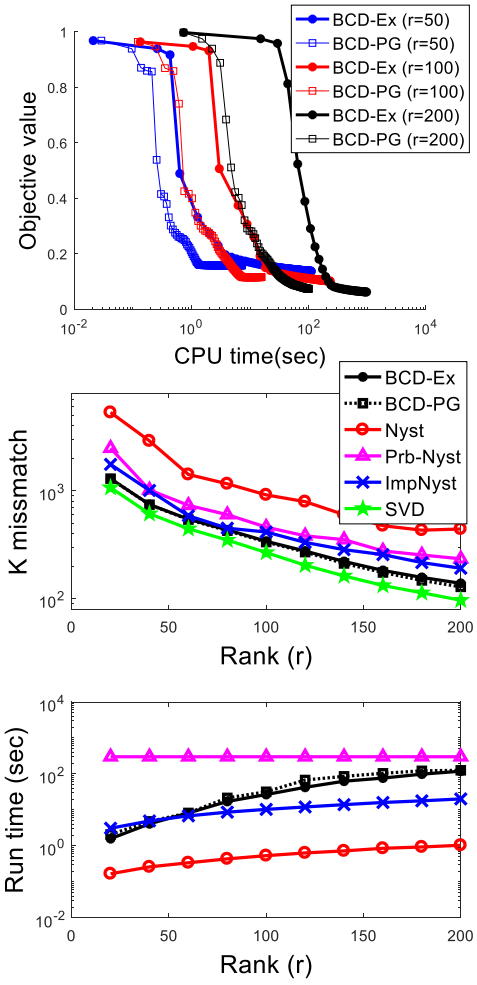
Adult Dataset

N=32,650, d= 123



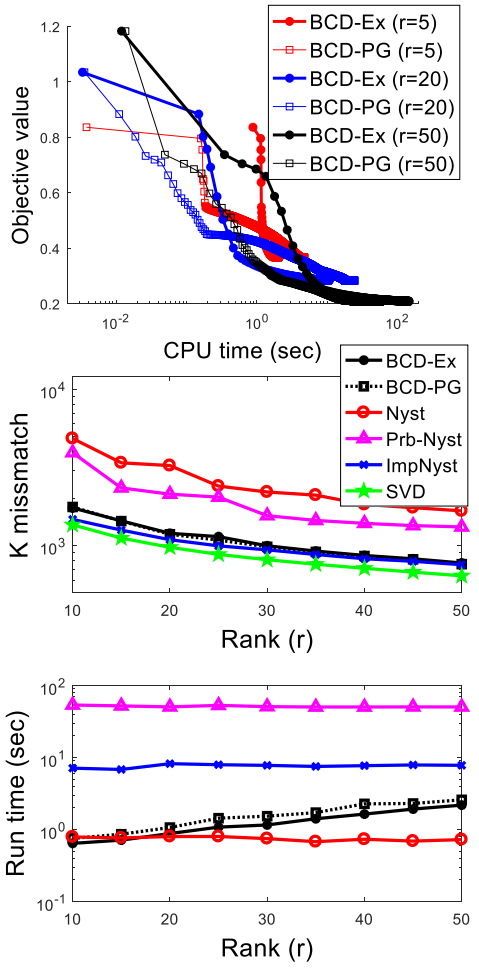
IJCNN Dataset

N=140,230 , d= 22



Slice Dataset

N=53,500 , d= 384



F. Sheikholeslami and G. B. Giannakis, "Scalable Kernel-based Learning via Low-rank Approximation of Lifted Data," *Proc. Allerton Conf.* 2017.

Datasets available at UCI repository: http://archive.ics.uci.edu/ml/datasets.html

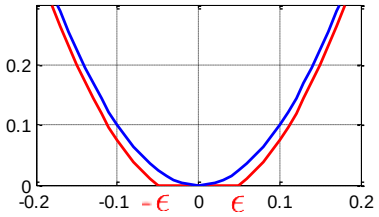
Online function approximation on a budget

- Low-rank (r) subspace tracking [Mardani-Mateos-GG'14] here on **lifted** data

$$\min_{\mathbf{A},\{\mathbf{q}_\nu\}_{\nu=1}^n} \frac{1}{n}\sum_{\nu=1}^{n} \underbrace{\|\boldsymbol{\phi}(\mathbf{x}_\nu)-\boldsymbol{\Phi}_n\mathbf{A}\mathbf{q}_\nu\|_{\mathcal{H}}^2}_{\ell_n(\mathbf{x}_\nu;\mathbf{A},\mathbf{q}_\nu;\mathbf{x}_{1:n}) :=} + \frac{\lambda}{2n}\underbrace{\left(\|\boldsymbol{\Phi}_n\mathbf{A}\mathbf{Q}_n^{1/2}\|_{*F}^2+\|\mathbf{Q}_n\|_F^2\right)}_{\text{Rank surrogate}}$$

$$k(\mathbf{x}_\nu,\mathbf{x}_\nu)-2\mathbf{k}_\nu^\top\mathbf{A}\mathbf{q}_\nu+\mathbf{q}_\nu^\top\mathbf{A}^\top\mathbf{K}_n\mathbf{A}\mathbf{q}_\nu$$

- Censoring can mitigate curse of dimensionality as $n\uparrow$



- Iteration $n-1$: $\mathbf{K}_{n-1}$ and $\mathbf{A}[n-1]$ available

S1. Find projection coefficients

$$\mathbf{q}_n=(\mathbf{A}^\top[n-1]\mathbf{K}_{n-1}\mathbf{A}[n-1]+\lambda\mathbf{I})^{-1}\mathbf{k}_{n-1}(\mathbf{x}_n)$$

S2. Update subspace factor (via stochastic gradient descent)

$$\mathbf{A}[n]=\min_{\mathbf{A}}\frac{1}{n}\sum_{\nu=1}^{n}\ell_n(\mathbf{x}_\nu;\mathbf{A},\mathbf{q}_\nu;\mathbf{x}_{1:n})+\frac{\lambda}{2n}\mathrm{Tr}\{\mathbf{A}^\top\mathbf{K}_n\mathbf{A}\}$$

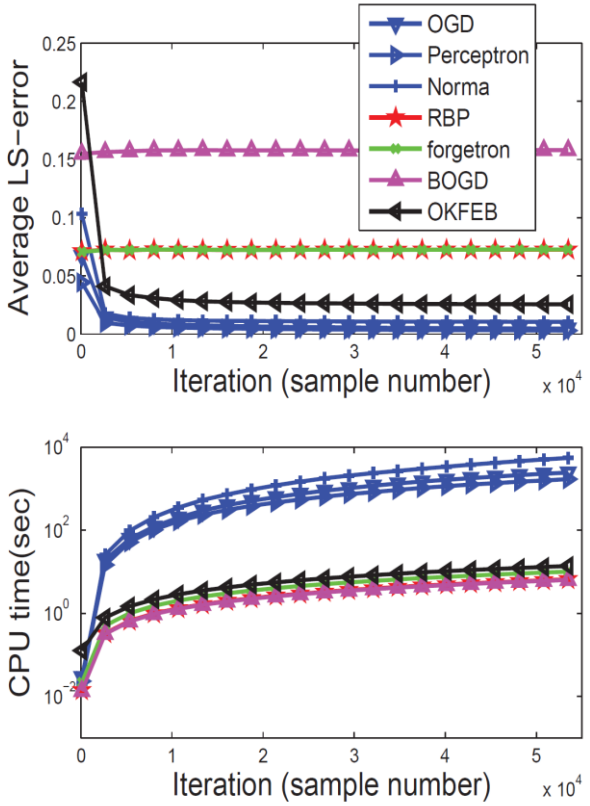
- If budget exceeded, remove the row of ***A*** with minimum ℓ_2-norm

Budget maintenance

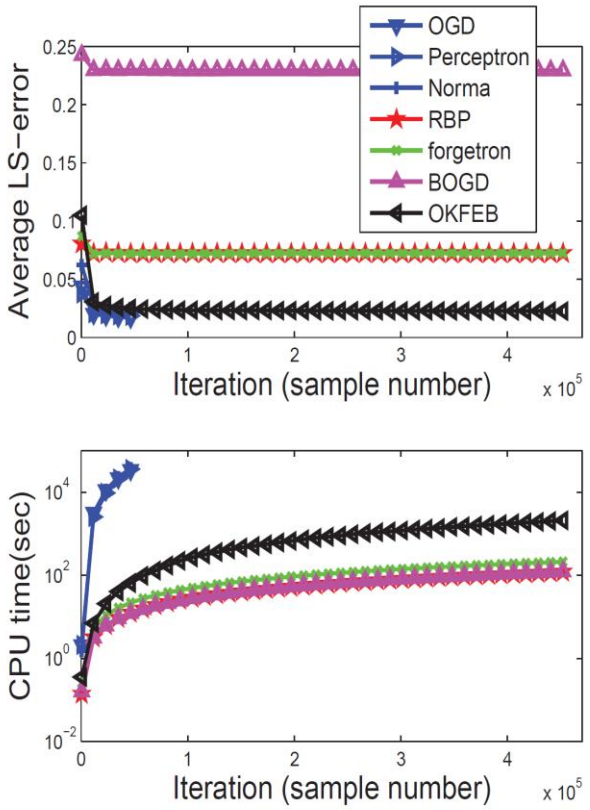
F. Sheikholeslami, D. K. Berberidis, and G. B. Giannakis, "Kernel-based Low-rank Feature Extraction on a Budget for Big Data Streams," arxiv:1601.07947.

OK-FEB with linear classification and regression

- Slice dataset (regression)
- N=53,500 , d= 384, r=10, B=15



- Year dataset (regression)
- N=463,700 , d= 90, r=10, B=15



✓ OK-FEB LSVM outperforms budgeted K-SVM/SVR variants in classification/regression

Tracking dynamic subspaces

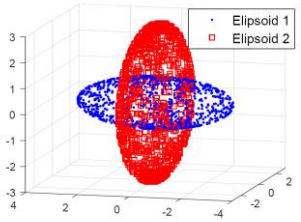
❑ Recency-aware removal rule

✓ **Recency factor**

- Associate factor r_i to the i-th SV in the budget
- Decay r_i with inclusion of a new SV, $r_i = \beta r_i \; 0 < \beta \leq 1$

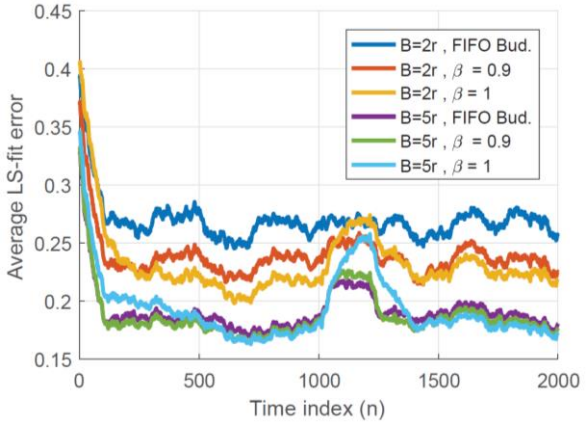
✓ **Removal rule**

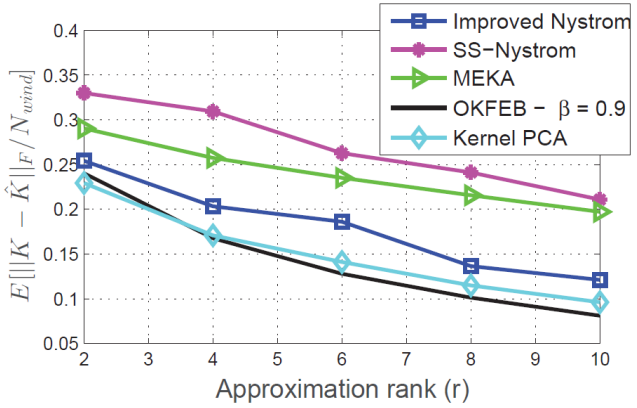
$$\hat{i}_* = \arg \min_{i=1,2,\ldots,B+1} r_i \|\mathbf{a}_i[n]\|_2$$



✓ Parameter β trades off tracking for precision

❑ Synthetic dataset $\{\mathbf{x}_t\}_{t=1}^{1000}$ and $\{\mathbf{x}_t\}_{t=1000}^{2000}$ drawn from two subspaces



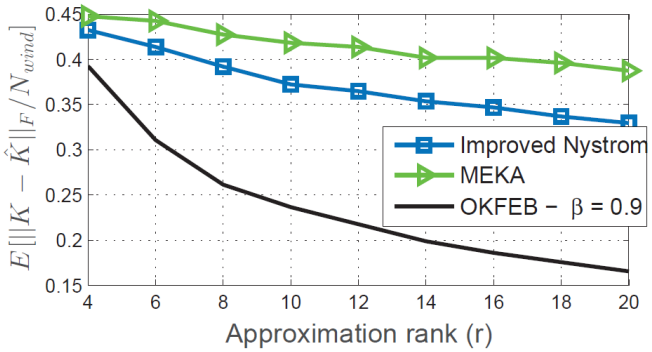


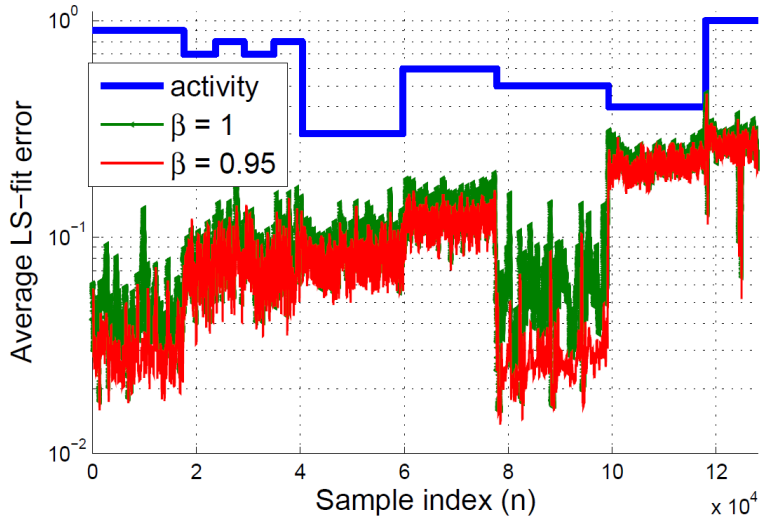
✓ Successful use of affordable budget for nonlinear subspace tracking

✓ Smaller values of β provide ***faster tracking***, while larger values increase ***fitting precision***

Online physical activity tracking

- Subjects asked to do various physical activities
- *d* =13 quantities measured from chest, ankle and wrist via wireless MUs
- Gaussian kernel; and $(r, B) = (10, 15)$





Average LS-fit

Code	Activity	$\beta = 1$	$\beta = 0.9$	FIFO Bud.
0.3	Walking	0.099 ±0.016	0.074 ±0.012	0.074 ±0.012
0.4	Running	0.227 ±0.025	0.187 ±0.022	0.187 ±0.022
0.5	Cycling	0.058 ±0.027	0.028 ±0.012	0.028 ±0.12
0.6	Nordic Walking	0.130 ±0.020	0.103 ±0.016	0.103 ±0.016
0.7	Ascending Stairs	0.079 ±0.022	0.063 ±0.018	0.063 ±0.018
0.8	Descending Stairs	0.094 ±0.021	0.066 ±0.016	0.065 ±0.016
0.9	Vacuum cleaning	0.045 ±0.013	0.029 ±0.008	0.029 ±0.008
1.0	Rope jumping	0.272 ±0.063	0.238 ±0.057	0.238 ±0.057

F. Sheikholeslami, D. K. Berberidis, and G. B. Giannakis, "Kernel-based Low-rank Feature Extraction on a Budget for Big Data Streams," arxiv:1601.07947.

Summary

- Kernel-based learning
  - Sparsity-aware low-rank approximation of lifted data
  - Nuclear norm regularization for online approximation
  - Budget enforcement for affordable memory and computation
  - Theoretical guarantees and test results
- Future directions
  - Nonlinear feature extraction for canonical correlation analysis
  - Kernel-based feature extraction over 2-D signals on networks

Thank you!