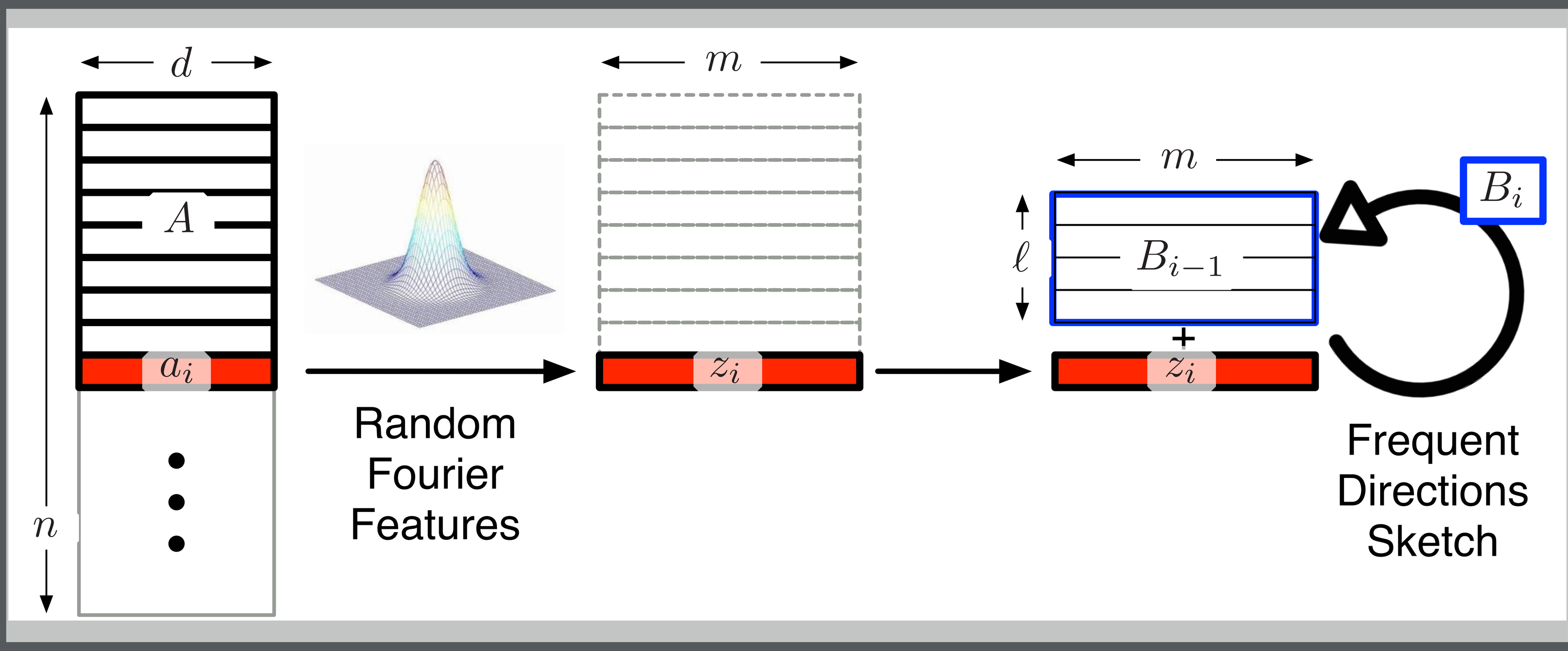
Streaming Kernel Principal Component Analysis

Mina Ghashami[1], Daniel J. Perry[2], Jeff M. Phillips[1]

School of Computing[1] and SCI Institute[2], University of Utah, USA

Objectives for a streaming KPCA

- Small space requirement
- Small training time (process training data)
- Small testing time (evaluate unseen test data)
- Bound on potential error



Algorithm: SKPCA

Input: Data $\mathbf{A} \in \mathbb{R}^{\mathbf{n}\times \mathbf{d}}$, kernel $\mathbf{K}$, $\ell, \mathbf{m} \in \mathbb{Z}^+$
Output: RFF maps $[\mathbf{f}_1, \cdots, \mathbf{f}_m]$, subspace $\mathbf{W}$
$[\mathbf{f}_1, \cdots, \mathbf{f}_m] = \mathrm{RFF}(\mathbf{K}, \mathbf{m})$
$\mathbf{B} \leftarrow \mathbf{0}^{\ell\times \mathbf{m}}$
for $\mathbf{i} \in [\mathbf{n}]$ **do**
  $\mathbf{z}_i = \sqrt{\frac{2}{m}}[\mathbf{f}_1(\mathbf{a}_i), \cdots, \mathbf{f}_m(\mathbf{a}_i)]$ } RFF projection
  $\mathbf{B} \leftarrow \mathbf{z}_i$
  if B has no zero valued rows **then**
   $[\mathbf{Y}, \mathbf{\Sigma}, \mathbf{W}] \leftarrow \mathrm{svd}(\mathbf{B})$
   $\mathbf{B} \leftarrow \sqrt{\max\{0, \mathbf{\Sigma}^2 - \mathbf{\Sigma}^2_{\ell/2,\ell/2}\mathbf{I}_\ell\}} \cdot \mathbf{W}^T$
  end if
end for
Return $[\mathbf{f}_1, \cdots, \mathbf{f}_m]$ and $\mathbf{W}$

Frequent Directions

Theorem 1: Spectral error bound

Let $\mathbf{G} = \mathbf{\Phi\Phi}^T$ be the exact kernel matrix over $\mathbf{n}$ points. Let $\tilde{\mathbf{G}} = \mathbf{ZW}^T\mathbf{WZ}^T$ be the result of $\mathbf{Z}$ from $\mathbf{m} = \mathbf{O}((1/\varepsilon^2)\log(\mathbf{n}/\delta))$ RFFand $\mathbf{W}$ from running Algorithm SKPCA with $\ell = 4/\varepsilon$. Then with probability at least $1 - \delta$, we have $\|\mathbf{G} - \tilde{\mathbf{G}}\|_2 \leq \varepsilon \mathbf{n}$.

Theorem 2: Frobenius error bound

Given that $\|\mathbf{G} - \mathbf{G}'\|_2 \leq \varepsilon\mathbf{n}$ we can bound $\|\mathbf{G} - \mathbf{G}'_k\|_F \leq \|\mathbf{G} - \mathbf{G}_k\|_F + \varepsilon\sqrt{\mathbf{kn}}$.

Runtime bounds to obtain $\|\mathbf{G}' - \mathbf{G}\|_2 \leq \varepsilon \mathbf{n}$

	Train time	Test time
KPCA	$\mathbf{O}(\mathbf{n}^2(\mathbf{n}+\mathbf{d}))$	$\mathbf{O}(\mathbf{n}(\mathbf{d}+\mathbf{n}^2))$
Nyström	$\mathbf{O}(\mathbf{nd}+\mathbf{n}/\varepsilon^2+1/\varepsilon^4)$	$\mathbf{O}(\mathbf{d}/\varepsilon^2+1/\varepsilon^4)$
RNCA	$\mathbf{O}(\mathbf{n}((\mathbf{d}/\varepsilon^2)\log \mathbf{n} + (1/\varepsilon^4)\log^2 \mathbf{n}))$	$\mathbf{O}((1/\varepsilon^2)(\mathbf{d}+\mathbf{n})\log \mathbf{n})$
SKPCA	$\mathbf{O}(\mathbf{n}\log \mathbf{n}(\mathbf{d}/\varepsilon^2+1/\varepsilon^3))$	$\mathbf{O}((\mathbf{d}+1/\varepsilon)/\varepsilon^2\log \mathbf{n})$

Space bounds to obtain $\|\mathbf{G}' - \mathbf{G}\|_2 \leq \varepsilon \mathbf{n}$

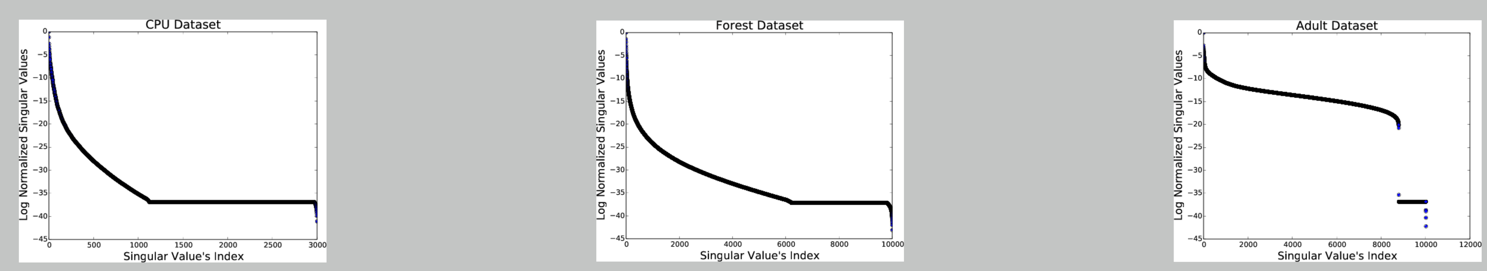
	Space
KPCA	$\mathbf{O}(\mathbf{n}^2+\mathbf{nd})$
Nyström	$\mathbf{O}(\mathbf{d}/\varepsilon^2+1/\varepsilon^4)$
RNCA	$\mathbf{O}((\mathbf{d}/\varepsilon^2)\mathbf{n}\log \mathbf{n})$
SKPCA	$\mathbf{O}(((\mathbf{d}+1/\varepsilon)/\varepsilon^2)\log \mathbf{n})$

Previous work

- Existing approaches to streaming/online KPCA either provide no error bound, require substantial space during training time, or have an expensive matrix inverse at test time.
- *Incremental KPCA techniques* update the eigenspace of kernel PCA without storing training data, but suffer from unbounded compound error in intermediate approximations of the eigenspace on adversarial data sets.
- *Nystrom approximation* methods approximate the kernel (Gram) matrix $\mathbf{G} = \mathbf{CW}_k^{\dagger}\mathbf{C}^T$, by sampling columns of $\mathbf{G}$ in a non-streaming setting, but require a costly matrix inverse at test time.
- *Randomized Nonlinear Component Analysis* (RNCA) uses a *Random Fourier Feature (RFF)* approximation to $\mathbf{G}$ via randomized feature maps by directly approximating the lifting function, but use an exact (costly) covariance computation.
- We propose *Streaming KPCA (SKPCA)*, combining the computational benefits of Random Fourier Features (RFF) and approximation bounds of Frequent Directions (FD) to achieve the stated goals.

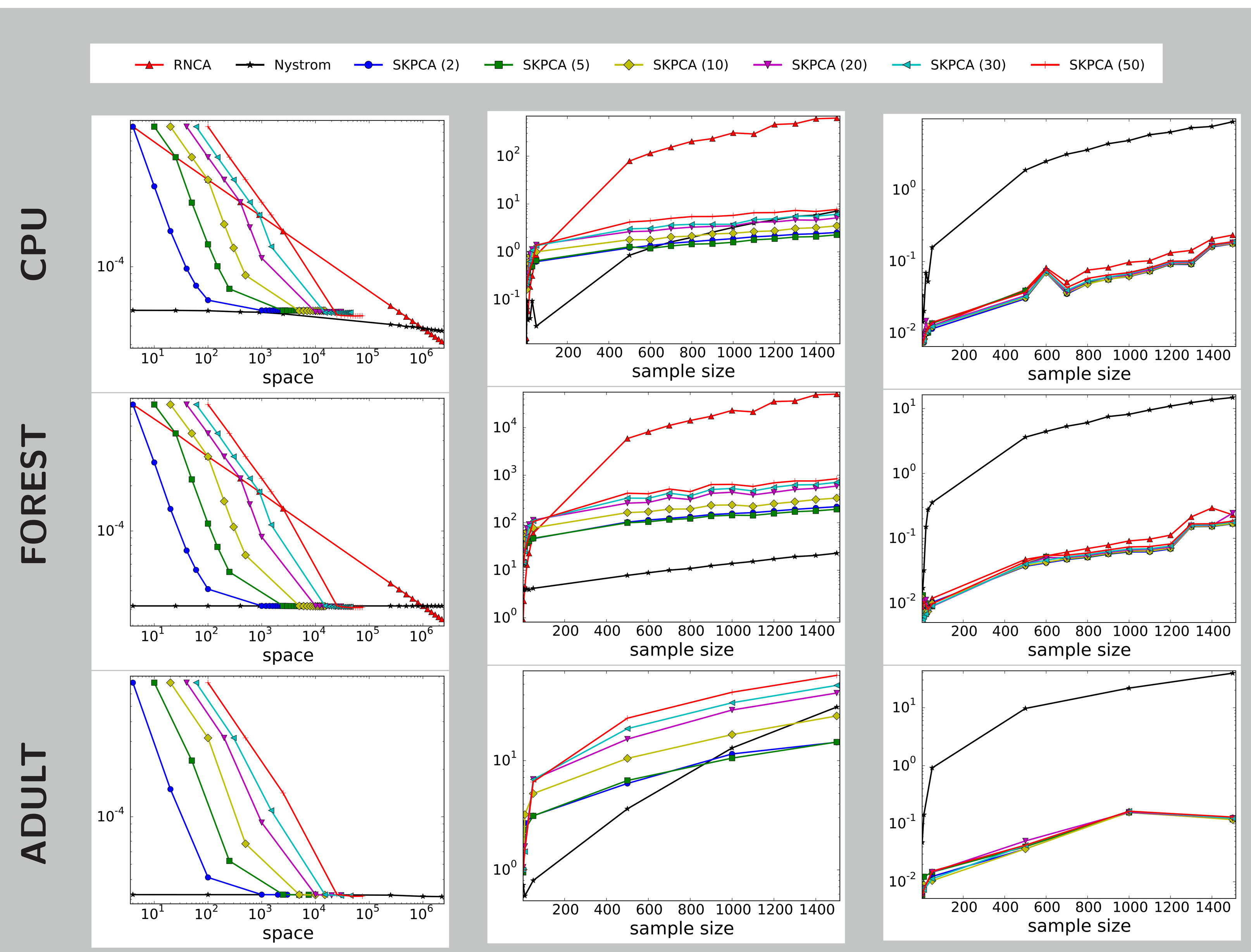
Datasets

- Methods were compared on real and synthetic datasets, including three real datasets below from the UCI machine learning repository.
- The kernel matrix was found using an RBF kernel (or RFF equivalent) with the bandwidth set to the averge inter-point distance – the spectra and input data sizes from the three datasets are shown below.



CPU	FOREST	ADULT
7373×21	523910×54	33561×123

Results



Frobenius error   Training time (sec)   Testing time (sec)

- *Frobenius error* is the approximation error, $\|\mathbf{G}' - \mathbf{G}\|_F$.
- The size of the sketch, ℓ, is a parameter and we compare several choices, $\ell = \{2, 5, 10, 20, 30, 50\}$, indicated by parenthesis in the legend.

Discussion

- *Nyström*: fast training time (random sampling), considerably slower testing time due to sample Gram matrix inversion.
- *RNCA*: fast testing time (matrix multiplication), training slower because complete covariance accumulation as data are observed
- *SKPCA*: obtains a more balanced runtime where both training and testing are competitive
  - Error is competitive with previous methods (all methods less than 10^{-3} in error)
  - Improved error vs space for RFF based methods

http://www.cs.utah.edu   NSF CCF-1350888, IIS-1251019, ACI-1443046, and CNS-1514520
{ghashami,dperry,jeffp}@cs.utah.edu