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 $A \in \mathbb{R}^{n \times d}$, kernel $K$, $\ell, m \in \mathbb{Z}^+$
Output: RFF maps $[f_1, \ldots, f_m]$, subspace $W$

1. $[f_1, \ldots, f_m] = \text{RFF}(K, m)$
2. $B \leftarrow 0^{m \times m}$
3. for $i \in [n]$ do
   - $z_i = \sqrt{2} f_i(a)$
   - $B \leftarrow B + z_i z_i^T$
   - if $B$ has no zero valued rows then
     - $[Y, \Sigma, W] \leftarrow \text{svd}(B)$
     - $B \leftarrow \max(0, \Sigma^2 - \Sigma^2_{r+1}, \Sigma^2_{r+1}) \cdot W^T$
4. end for
5. Return $[f_1, \ldots, f_m]$ and $W$

Theorem 1: Spectral error bound

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

Theorem 2: Frobenius error bound

Given that $\|G - G'\|_2 \leq \epsilon n$ we can bound $\|G - G'_K\|_F \leq \|G - G_K\|_F + \epsilon \sqrt{k} n$.

Runtime bounds to obtain $\|G' - G\|_2 \leq \epsilon n$

<table>
<thead>
<tr>
<th>Method</th>
<th>Train time</th>
<th>Test time</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPCA</td>
<td>$O(n^2 + nd)$</td>
<td>$O(n^2 + nd)$</td>
</tr>
<tr>
<td>Nyström</td>
<td>$O(n \log n + (1/\epsilon^2) \log^2 n)$</td>
<td>$O(d/\epsilon^2 + 1/\epsilon^4)$</td>
</tr>
<tr>
<td>RNCA</td>
<td>$O(n \log n (d/\epsilon^2 + 1/\epsilon^4))$</td>
<td>$O((d+1/\epsilon)/\epsilon^2 \log n)$</td>
</tr>
<tr>
<td>SKPCA</td>
<td>$O(n \log n (d/\epsilon^2 + 1/\epsilon^4))$</td>
<td>$O((d+1/\epsilon)/\epsilon^2 \log n)$</td>
</tr>
</tbody>
</table>

Space bounds to obtain $\|G' - G\|_2 \leq \epsilon n$

<table>
<thead>
<tr>
<th>Method</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPCA</td>
<td>$O(n^2 + nd)$</td>
</tr>
<tr>
<td>Nyström</td>
<td>$O(d/\epsilon^2 + 1/\epsilon^4)$</td>
</tr>
<tr>
<td>RNCA</td>
<td>$O(n \log n (d/\epsilon^2 + 1/\epsilon^4))$</td>
</tr>
<tr>
<td>SKPCA</td>
<td>$O((d+1/\epsilon)/\epsilon^2 \log n)$</td>
</tr>
</tbody>
</table>

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.
- Nyström approximation methods approximate the kernel (Gram) matrix $G = CW[C^T]$, by sampling columns of $G$ 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 $G$ 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 average inter-point distance – the spectra and input data sizes from the three datasets are shown below.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>7373 x 21</td>
</tr>
<tr>
<td>FOREST</td>
<td>523910 x 54</td>
</tr>
<tr>
<td>ADULT</td>
<td>33561 x 123</td>
</tr>
</tbody>
</table>

Results

- CPU: fastest test time, considerable slower training time due to sample Gram matrix inversion.
- RNCA: fast testing time (matrix multiplication), training slower because complete covariance accumulation as data are observed.
- SKPCA: obtained 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.

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.