Low Rank Matrix Approximation
PRESENTED BY Edo Liberty - April 24, 2015

Collaborators: Nir Ailon, Steven Zucker, Zohar Karnin, Dimitris Achlioptas, Per-Gunnar Martinsson, Vladimir Rokhlin, Mark Tygert, Christos Boutsidis, Franco Woolfe, Maxim Sviridenko, Dan Garber, Yoelle Maarek, Amit Singer, David Woodruff, Jeff Phillips, Mina Ghashami, Huan Wang, Daniel Hsu
Often our data is represented by a large matrix.
Matrix Data

We will think of \( A \in \mathbb{R}^{d \times n} \) as \( n \) column vectors in \( \mathbb{R}^d \) and typically \( n \gg d \).

Typical web scale data:

<table>
<thead>
<tr>
<th>Data</th>
<th>Columns</th>
<th>Rows</th>
<th>( d )</th>
<th>( n )</th>
<th>sparse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Textual</td>
<td>Documents</td>
<td>Words</td>
<td>( 10^5 - 10^7 )</td>
<td>( &gt; 10^{10} )</td>
<td>yes</td>
</tr>
<tr>
<td>Actions</td>
<td>Users</td>
<td>Types</td>
<td>( 10^1 - 10^4 )</td>
<td>( &gt; 10^7 )</td>
<td>yes</td>
</tr>
<tr>
<td>Visual</td>
<td>Images</td>
<td>Pixels, SIFT</td>
<td>( 10^5 - 10^6 )</td>
<td>( &gt; 10^8 )</td>
<td>no</td>
</tr>
<tr>
<td>Audio</td>
<td>Songs, tracks</td>
<td>Frequencies</td>
<td>( 10^5 - 10^6 )</td>
<td>( &gt; 10^8 )</td>
<td>no</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Examples</td>
<td>Features</td>
<td>( 10^2 - 10^4 )</td>
<td>( &gt; 10^6 )</td>
<td>yes/no</td>
</tr>
<tr>
<td>Financial</td>
<td>Prices</td>
<td>Items, Stocks</td>
<td>( 10^3 - 10^5 )</td>
<td>( &gt; 10^6 )</td>
<td>no</td>
</tr>
</tbody>
</table>
Matrix Data

Low rank matrix approximation is helpful for

- Dimension reduction
- Signal processing
- Compression
- Classification
- Regression
- Clustering
- …
Singular Value Decomposition (SVD)

\[ A = U \Sigma V^T \]

\[ U \in \mathbb{R}^{d \times d} \quad \Sigma \in \mathbb{R}^{d \times d} \quad V^T \in \mathbb{R}^{d \times n} \]

\[ U^T U = I_d \quad \Sigma_{1,1} \geq \cdots \geq \Sigma_{d,d} \geq 0 \]

\[ V^T V = I_d \]
Best rank $k$ Approximation

$$A_k = U_k \Sigma_k V_k^T$$

$$U_k \in \mathbb{R}^{d \times k} \quad \Sigma_k \in \mathbb{R}^{k \times k} \quad V_k^T \in \mathbb{R}^{k \times n}$$

$$U_k^T U_k = I_k \quad \Sigma_{1,1} \geq \ldots \geq \Sigma_{k,k} \geq 0 \quad V_k^T V_k = I_k$$

$B = A_k$ minimizes $\|A - B\|_2$ and $\|A - B\|_F$ among all rank $k$ matrices.
### Best rank $k$ Approximation

Block power methods and Lanczos like methods:

- $\tilde{O}(k)$ passes over the matrix.
- $\tilde{O}(ndk)$ operations

$\tilde{O}(\cdot)$ hides logarithmic factors and spectral gap dependencies.

By first computing $AA^T$

- $\Omega(d^2)$ space
- $O(nd^2)$ operations

Assuming $d = o(n)$ and naive matrix matrix multiplication.
Matrix Approximation

Let $P_k^A = U_k U_k^T$ be the best rank $k$ projection of the columns of $A$

$$\|A - P_k^A A\|_2 = \|A - A_k\|_2 = \sigma_{k+1}$$

Let $P_k^B$ be the best rank $k$ projection for $B$

$$\|A - P_k^B A\|_2 \leq \sigma_{k+1} + \sqrt{2\|AA^T - BB^T\|}$$  \[FKV04\]

From this point on, our goal is to find $B$ which is:

1. $\|AA^T - BB^T\| \leq \varepsilon\|AA^T\|$
2. Small, $B \in \mathbb{R}^{d \times \ell}$ and $\ell \ll d$
3. Computationally easy to obtain from $A$
Random projection based algorithms
Random projection

1. Output $B = AR$

2. Where $R \in \mathbb{R}^{n \times \ell}$ such that $R_{i,j} \sim \mathcal{N}(0, 1/\ell)$.

$\mathbb{E}[RR^T] = I_n$
Random projection

1. Output $B = AR$

2. Where $R \in \mathbb{R}^{n \times \ell}$ such that $R_{i,j} \sim \mathcal{N}(0, 1/\ell)$.

Random projection

Johnson-Lindenstrauss property [JL84, FM87, DG99]

The matrix $R$ exhibits the Johnson-Lindenstrauss property. For any $y \in \mathbb{R}^n$

$$
\Pr \left( \left| \|yR\|^2 - \|y\|^2 \right| > \varepsilon \|y\|^2 \right) \leq e^{-c\ell \varepsilon^2}
$$

If $\ell = \tilde{O}(\text{Rank}(A)/\varepsilon^2)$ then by the union bound we have

$$
\|A^T A - B^T B\| = \sup_{\|x\|=1} \left| \|xA\|^2 - \|xAR\|^2 \right| \leq \varepsilon \|AA^T\|
$$

This gives us exactly what we need!

Random projection

- 1 pass
- $O(nd\ell)$ operations
Fast random projection

- This can be accelerated by making $R$ sparser [Ach03, Mat08, DKS10, KN10].
- But in general, $R$ cannot be “much sparser” [KN10, NNW12, NN13, NN14].

Faster Johnson-Lindenstrauss transforms require very different machinery [AC06, AL09, AC10, LAS11, AL11, KW11, AR14].

- 1 pass (by row)
- $O(nd \log(\ell))$ operations
Matrix approximation in the streaming setting
Data is dynamically aggregated

Sometimes we get one column at a time (row operations impossible...)

How many photos are uploaded to Flickr every day, month, year?
Data is dynamically aggregated

Sometimes, we cannot even store the entire matrix.
Streaming Matrices

Note that $AA^T$ can be trivially computed from the stream of columns $A_i$

$$AA^T = \sum_{i=1}^{n} A_i A_i^T$$

In words, $AA^T$ is the sum of outer products of the columns of $A$.
Streaming Matrices

Note that $AA^T$ can be trivially computed from the stream of columns $A_i$

$$AA^T = \sum_{i=1}^{n} A_i A_i^T$$

In words, $AA^T$ is the sum of outer products of the columns of $A$.

Naïve solution

Compute $AA^T$ in time $O(nd^2)$ and space $O(d^2)$. Compute the best rank-k projection for $AA^T$ in $o(nd^2)$.

Which is hopeless when $d$ is large!
Column Sampling

Sample only $\ell$ columns where $[FKV04, AW02, Ver, DK03, RV07, Oli10]$

$$\ell \in O\left(\frac{r \log(r)}{\varepsilon^2}\right)$$

Each column of $B$ is $A^{(j)}/p_j$ w.p. $p_j = \|A^{(j)}\|_2^2 / \|A\|_F^2$

**Column sampling based on $\ell_2^2$ norm**

1. It can be performed in a column stream, $O(nnz(A))$ operations
2. The result is sparse if the data is sparse, potentially $o(dl)$ space

$$r = \|A\|_F^2 / \|A\|_2^2$$ is the numeric/stable rank of $A$. 

Feature Hashing

Use the “count-sketch” matrix $R$ that contains one $\{-1, 1\}$ per row.

1. It can be applied in streaming in $O(nnz(A))$ operations
2. The result is dense, $\Omega(d\ell)$ space
3. Has some other surprising properties...

[CCFC02, WDL+09, CW12]
Experiments

The error term $\|AA^T - BB^T\|$ reduces like $1/\sqrt{\ell}$. 
Frequent Directions
Frequent Directions

Lemma from \[\text{Lib13}\]

One can maintain a matrix $B$ with only $\ell = O(1/\varepsilon)$ columns s.t.

$$\|AA^T - BB^T\|_2 \leq \varepsilon\|A\|_F^2$$

Intuition:

Extend Frequent-items \[\text{MG82, DLOM02, KPS03, MAEA05}\]
Frequent Items

Obtain the frequency $f(i)$ of each item in the stream of items
Frequent Items

With \( d \) counters it’s easy but not good enough (IP addresses, queries....)
(Misra-Gries) Let's keep less than a fixed number of counters $\ell$. 
If an item has a counter we add 1 to that counter.
Frequent Items

Otherwise, we create a new counter for it and set it to 1
But now we do not have less than $\ell$ counters.
Frequent Items

Let $\delta$ be the median counter value at time $t$

\[ \delta = f_{\ell/2} = 2 \]
Frequent Items

Decrease all counters by $\delta$ (or set to zero if less than $\delta$)
Frequent Items

And continue...
Frequent Items

The approximated counts are $f'$

\[ f'(\text{green}) = 1 \]
\[ f'(\text{purple}) = 0 \]
Frequent Items

- We increase the count by only 1 for each item appearance.

\[ f'(i) \leq f(i) \]

- Because we decrease each counter by at most \( \delta_t \) at time \( t \)

\[ f'(i) \geq f(i) - \sum_t \delta_t \]

- Calculating the total approximated frequencies:

\[ 0 \leq \sum_i f'(i) \leq \sum_t 1 - (\ell/2) \cdot \delta_t = n - (\ell/2) \cdot \sum_t \delta_t \]

\[ \sum_t \delta_t \leq 2n/\ell \]

- Setting \( \ell = 2/\varepsilon \) yields

\[ |f(i) - f'(i)| \leq \varepsilon n \]
Email threading

Find all email pairs such that $\Pr(e_1|e_2) \geq \theta$ [AKLM13].
Frequent Directions

We keep a sketch of at most $\ell$ columns
Frequent Directions

We maintain the invariant that some columns are empty (zero valued)
Frequent Directions

Input vectors are simply stored in empty columns
Frequent Directions

Input vectors are simply stored in empty columns
Frequent Directions

When the sketch is ‘full’ we need to zero out some columns...
Frequent Directions

Using the SVD we compute $B = USV^T$ and set $B_{new} = US$. 

$B = USV^T$

$B_{new} = US$
Frequent Directions

\[ B = U S V^T \]

\[ B_{new} = U S \]

\[ V^T \]

Note that \( B B^T = B_{new} B_{new}^T \) so we don’t “lose” anything.
Frequent Directions

The columns of $B$ are now orthogonal and in decreasing magnitude order.
Frequent Directions

Let $\delta = \|B_{\ell/2}\|^2$
Frequent Directions

Reduce column $\ell_2^2$-norms by $\delta$ (or nullify if less than $\delta$)
Frequent Directions

Start aggregating columns again...
Frequent Directions

Input: $\ell, \ A \in \mathbb{R}^{d \times n}$

$B \leftarrow$ all zeros matrix $\in \mathbb{R}^{d \times \ell}$

for $i \in [n]$ do

Insert $A_i$ into a zero valued column of $B$

if $B$ has no zero valued columns then

$[U, \Sigma, V] \leftarrow \text{SVD}(B)$

$\delta \leftarrow \sigma_{\ell/2}^2$

$\tilde{\Sigma} \leftarrow \sqrt{\max(\Sigma^2 - I_\ell \delta, 0)}$

$B \leftarrow U \tilde{\Sigma}$

# At least half the columns of $B$ are zero.

Return: $B$
Bounding the error

We first bound $\|AA^T - BB^T\|$

$$\sup_{\|x\| = 1} \| xA \|^2 - \| xB \|^2 = \sup_{\|x\| = 1} \sum_{t=1}^{n} [\langle x, A_t \rangle^2 + \| xB_t^{-1} \|^2 - \| xB_t \|^2]$$

$$= \sup_{\|x\| = 1} \sum_{t=1}^{n} [\| xC_t \|^2 - \| xB_t \|^2]$$

$$\leq \sum_{t=1}^{n} \| C_t^T C_t - B_t^T B_t \| \cdot \| x \|^2 \leq \sum_{t=1}^{n} \delta_t$$

Which gives:

$$\|AA^T - BB^T\| \leq \sum_{t=1}^{n} \delta_t$$
Bounding the error

We compute the Frobenius norm of the final sketch.

\[ 0 \leq \| B \|_F^2 = \sum_{t=1}^{n} [\| B_t \|_F^2 - \| B_t-1 \|_F^2] \]

\[ = \sum_{t=1}^{n} [(\| C_t \|_F^2 - \| B_t-1 \|_F^2) - (\| C_t \|_F^2 - \| B_t \|_F^2)] \]

\[ = \sum_{t=1}^{n} \| A_t \|^2 - tr (C^T C_t - B^T B_t) \leq \| A \|_F^2 - (\ell/2) \sum_{t=1}^{n} \delta_t \]

Which gives:

\[ \sum_{t=1}^{n} \delta_t \leq 2 \| A \|_F^2 / \ell \]
Bounding the error

We saw that:

$$\|AA^T - BB^T\| \leq \sum_{t=1}^{n} \delta_t$$

and that:

$$\sum_{t=1}^{n} \delta_t \leq 2\|A\|_F^2/\ell$$

Setting $\ell = 2/\varepsilon$ yields

$$\|AA^T - BB^T\| \leq \varepsilon\|A\|_F^2.$$ 

The two proofs are very similar...
## Stronger bounds

**Lemma: covariance approximation guarantee** \[\text{[GP14, GLPW15]}\]

\[
\|A^T A - B^T B\|_2 \leq \|A - A_k\|_F^2 / (\ell - k) \text{ for any } k < \ell.
\]

**Lemma: projection approximation guarantee** \[\text{[GP14]}\]

\[
\|A - \pi_{B_k}(A)\|_F^2 \leq (1 + \frac{k}{\ell - k}) \|A - A_k\|_F^2 \text{ for any } k < \ell.
\]

**Lemma: space optimality** \[\text{[Woo14, GLPW15]}\]

Frequent directions is space optimal. Any algorithm (randomized or not) with matching guarantees must require as much space, up to a word-size factor.
Frequent Directions

Slower than hashing or sampling but still very fast.
Frequent Directions

The error term $\|AA^T - BB^T\|$ reduces like $1/\ell$
Matrix approximation online
Online PCA

Consider clustering the reduced dimensional vectors online (e.g. [Mey01, LSS14])

The PCA algorithm must output $y_t$ before receiving $x_{t+1}$.
Online PCA, prior models

**Regret minimization:** Minimizes $\sum_t \|x_t - P_{t-1}x_t\|^2$ where $P_{t-1}$ is committed to before receiving $x_t$ [WK06, NKW13]

**Random projection:** Can guarantee online that $\| (X - (XY^+)Y \|_F^2$ is small [CW09, Sar06]

**Stochastic model:** Assumes $x_t$ are drawn i.i.d. from an unknown distribution [OK85, ACS13, MCJ13, BDF13]
Principal Component Analysis

Given a set of vectors $x_t \in \mathbb{R}^d$ the goal is to map them to $y_t \in \mathbb{R}^k$ that minimize:

$$\min_{\{\Phi | \Phi^T\Phi = I_k\}} \sum_i \|x_t - \Phi y_t\|_2^2$$
Online regression

Note that this is non trivial even when $d = 2$ and $k = 1$.

For $x_1$ there aren’t many options...
Online regression

Note that this is non trivial even when $d = 2$ and $k = 1$.

For $x_2$ this is already a non standard optimization problem
Online regression

Note that this is non trivial even when $d = 2$ and $k = 1$.

In general, the mapping $x_i \mapsto y_i$ is not necessarily linear.
### Lemma: online PCA with Frobenius bounds

\[ \min_{\Phi} \| X - \Phi Y \|^2_F \leq \| X_k \|^2_F + \epsilon \| X \|^2_F \]

with target dimension \( \ell \in O(k/\epsilon^2) \)

---

### Lemma: improved online PCA with spectral bounds

\[ \min_{\Phi} \| X - \Phi Y \|_2^2 \leq \sigma_{k+1}^2 + \epsilon \sigma_1^2 \]

with target dimension \( \ell = \tilde{O}(\frac{k}{\epsilon^2}) \)
Online PCA algorithm intuition

The covariance matrix $X^T X$ visualized as an ellipse.

$X^T X$
Online PCA algorithm intuition

The optimal residual is $R = X - X_k$
Online PCA algorithm intuition

Any residual $R = X - \Phi Y$ such that $\|R^T R\| \leq \sigma^2_{k+1} + \varepsilon \sigma^2_1$ would work
Online PCA algorithm intuition

Let us assume we know \( \Delta = \sigma_{k+1}^2 + \varepsilon \sigma_1^2 \).
Online PCA algorithm intuition

We start with mapping $x_t \mapsto 0$ and $R_{[1:t]} = X_{[1:t]}$
Online PCA algorithm intuition

This is continued as long as $\|R^T R\| \leq \Delta$
Online PCA algorithm intuition

When $\|R^T R\| > \Delta$ we update the projection to prevent it from happening.
We commit to a new online PCA direction $u_i$ such that $\|R^T R\| \leq \Delta$ again.
Online PCA with Spectral Bounds

input: $X$
$U \leftarrow$ all zeros matrix

for $x_t \in X$ do
    if $\| (I - UU^T)X_{1:t} \|_2^2 \geq \sigma_{k+1}^2 + \varepsilon_1^2$
        Add the top left singular vector of $(I - UU^T)X_{1:t}$ to $U$

yield $y_t = U^T x_t$
Open questions

- Reduce running time of Frequent Directions (there is some progress on that)
- Reduce running time of online PCA
- Reduce target dimension of online PCA, is it possible?
- Can we avoid the doubling trick in online PCA if we allow *scaled* isometric reconstructions?
Thank you!
Nir Ailon and Bernard Chazelle.
Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform.

Nir Ailon and Bernard Chazelle.
Faster dimension reduction.

Dimitris Achlioptas.

Raman Arora, Andy Cotter, and Nati Srebro.
Stochastic optimization of pca with capped msg.

Nir Ailon, Zohar Shay Karnin, Edo Liberty, and Yoelle Maarek.
Threading machine generated email.

Nir Ailon and Edo Liberty.
Fast dimension reduction using rademacher series on dual bch codes.

Nir Ailon and Edo Liberty.
An almost optimal unrestricted fast johnson-lindenstrauss transform.
Nir Ailon and Holger Rauhut.
Fast and rip-optimal transforms.

Rudolf Ahlswede and Andreas Winter.
Strong converse for identification via quantum channels.

Akshay Balsubramani, Sanjoy Dasgupta, and Yoav Freund.
The fast convergence of incremental pca.

Christos Boutsidis, Dan Garber, Zohar Shay Karnin, and Edo Liberty.
Online principal components analysis.

Moses Charikar, Kevin Chen, and Martin Farach-Colton.
Finding frequent items in data streams.

Kenneth L Clarkson and David P Woodruff.
Numerical linear algebra in the streaming model.

Kenneth L. Clarkson and David P. Woodruff.
Low rank approximation and regression in input sparsity time.
S. DasGupta and A. Gupta.
An elementary proof of the Johnson-Lindenstrauss lemma.

Petros Drineas and Ravi Kannan.

Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós.
A sparse johnson: Lindenstrauss transform.

Erik D. Demaine, Alejandro López-Ortiz, and J. Ian Munro.
Frequency estimation of internet packet streams with limited space.
Springer-Verlag.

Alan Frieze, Ravi Kannan, and Santosh Vempala.
Fast monte-carlo algorithms for finding low-rank approximations.

P. Frankl and H. Maehara.
The Johnson-Lindenstrauss lemma and the sphericity of some graphs.

Mina Ghashami, Edo Liberty, Jeff M. Phillips, and David P. Woodruff.
Frequent directions : Simple and deterministic matrix sketching.
Mina Ghashami and Jeff M. Phillips.  
Relative errors for deterministic low-rank matrix approximations. 

W. B. Johnson and J. Lindenstrauss.  
Extensions of Lipschitz mappings into a Hilbert space. 

Zohar Shay Karnin and Edo Liberty.  
Online pca with spectral bounds. 
*In progress*, 2015.

Daniel M. Kane and Jelani Nelson.  
A derandomized sparse johnson-lindenstrauss transform. 

A simple algorithm for finding frequent elements in streams and bags. 

Felix Krahmer and Rachel Ward.  
New and improved johnson-lindenstrauss embeddings via the restricted isometry property. 

Edo Liberty, Nir Ailon, and Amit Singer.  
Dense fast random projections and lean walsh transforms. 
Edo Liberty.
Simple and deterministic matrix sketching.
In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’13, pages 581–588, New York, NY, USA, 2013. ACM.

Edo Liberty, Ram Sriharsha, and Maxim Sviridenko.
An algorithm for online k-means clustering.

Ahmed Metwally, Divyakant Agrawal, and Amr El Abbadi.
Efficient computation of frequent and top-k elements in data streams.

Jirí Matousek.
On variants of the johnson-lindenstrauss lemma.

Ioannis Mitliagkas, Constantine Caramanis, and Prateek Jain.
Memory limited, streaming pca.

Adam Meyerson.
Online facility location.

Jayadev Misra and David Gries.
Finding repeated elements.
Jiazhong Nie, Wojciech Kotlowski, and Manfred K. Warmuth.
Online pca with optimal regrets.

Jelani Nelson and Huy L. Nguyen.
Sparsity lower bounds for dimensionality reducing maps.

Jelani Nelson and Huy L. Nguyên.
Lower bounds for oblivious subspace embeddings.

Jelani Nelson, Huy L. Nguyên, and David P. Woodruff.
On deterministic sketching and streaming for sparse recovery and norm estimation.

Erkki Oja and Juha Karhunen.
On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix.

Roberto Imbuzeiro Oliveira.
Sums of random hermitian matrices and an inequality by rudelson.

Mark Rudelson and Roman Vershynin.
Sampling from large matrices: An approach through geometric functional analysis.
Tamas Sarlos.
Improved approximation algorithms for large matrices via random projections.

Roman Vershynin.
A note on sums of independent random matrices afterahlswede-winter.
Lecture Notes.

Kilian Weinberger, Anirban Dasgupta, John Langford, Alex Smola, and Josh Attenberg.
Feature hashing for large scale multitask learning.

Manfred K. Warmuth and Dima Kuzmin.
Randomized PCA algorithms with regret bounds that are logarithmic in the dimension.

Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert.
A fast randomized algorithm for the approximation of matrices.

David P. Woodruff.
Low rank approximation lower bounds in row-update streams.