Low Rank Factorization Using Error Correcting Codes

James Folberth and Jessica Gronski

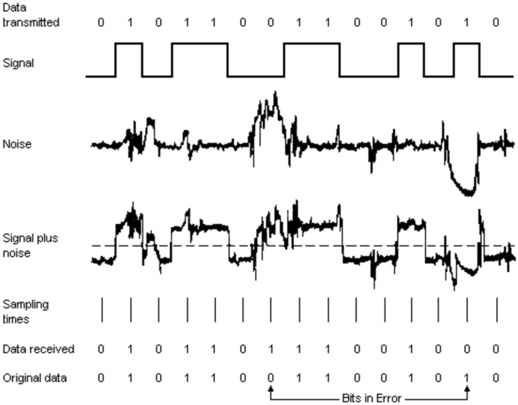
27 April, 2016

Brief Overview

- Randomization techniques for matrix approximations aim to compute basis that approximately spans the range of an $m \times n$ input matrix A.
- Form matrix-matrix product $Y = A\Omega$, Ω is $n \times \ell$ random matrix, $\ell \ll \{m, n\}$.
- Compute orthogonal basis, $Y = QR$, that identifies the range of reduced matrix Y.
- $A \approx QQ^T A$

Error Correcting Codes

- Data are transmitted from a source (transmitter) to a destination (receiver) through physical channels.

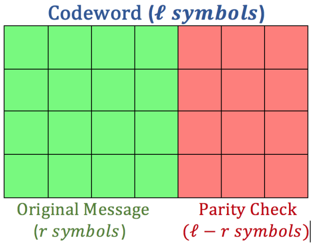


- Block of information encoded into binary vector, called *codeword*.
- Error correcting codes check correctness of the codeword received.

- Set of codewords corresponding to a set of data vectors that can possible be transmitted is called the *code*.

- A code is said to be linear when adding two codewords of the code component-wise modulo-2 arithmetic results in a third codeword of the code.

- A linear code C can be represented by:
  - codeword length ℓ
  - message length r



- To encode r bits, we require 2^r unique and well separated codewords.

- BCH codes form a class of cyclic error-correcting codes.
- Linear code C of length n is a *cyclic code* if it is invariant under a cyclic shift:
$$\mathbf{c} = (c_0, c_1, \ldots, c_{n-2}, c_{n-1}) \in C$$
if and only if
$$\tilde{\mathbf{c}} = (c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C,$$
[**?**].
- For any integers q and t, with t "small", a BCH code over $\mathbf{GF}(2) \sim \mathbb{Z}/2\mathbb{Z}$ has length $\ell = 2^q - 1$ and dimension $r = 2^q - 1 - tq$.
- Any two codewords maintain a minimum Hamming distance of $2t + 1$.

Connection to Randomized Methods

- In class, we've discussed randomized methods including the subsampled random Fourier Transform (SRFT).
- Ubaru et al. [**?**] propose using error correcting codes to find low rank approximations in a manner similar to SRFT.

- Let A be an $m \times n$ matrix with approximate rank k.
- **Goal:** Construct a lower dimensional subsampling matrix Ω so that $Y = A\Omega$ provides “good" approximation for range of A while STILL preserving the geometry of A, i.e. distances are preserved.

Subsampled Code Matrix

- Let A be an $m \times n$ matrix with approximate rank k.
- **Goal:** Construct a lower dimensional subsampling matrix Ω so that $Y = A\Omega$ provides "good" approximation for range of A while STILL preserving the geometry of A, i.e. distances are preserved.
- Choose the length of message $r \geq \lceil \log_2(n) \rceil$ and length of the code $\ell > k$, the target rank.

Subsampled Code Matrix

- Form the Subsampled Code Matrix (SCM) as:

$$\Omega_{n\times\ell} = \sqrt{\frac{2^r}{\ell}} D_{n\times n} S_{n\times 2^r} \Phi_{2^r\times\ell}$$

where

  - D is a random $n \times n$ diagonal matrix whose entries are independent random signs, i.e. random variables uniformly distributed on $\{\pm 1\}$.
  - S is a uniformly random downsampler, an $n \times 2^r$ matrix whose n rows are randomly selected from a $2^r \times 2^r$ identity matrix.
  - Φ is the $2^r \times \ell$ code matrix, generated using an $[\ell, r]$- linear coding scheme, with *binary phase-shift keying* mapping and scaled by $2^{-r/2}$ such that all columns have unit norm.

Subsampled Code Matrix

- Form the Subsampling Code Matrix as:

$$\Omega_{n\times\ell} = \sqrt{\frac{2^r}{\ell}} D_{n\times n} S_{n\times 2^r} \Phi_{2^r\times\ell}$$

  where

  - D is a random $n \times n$ diagonal matrix whose entries are independent random signs, i.e. random variables uniformly distributed on $\{\pm 1\}$.
  - S is a uniformly random downsampler, an $n \times 2^r$ matrix whose n rows are randomly selected from a $2^r \times 2^r$ identity matrix.
  - Φ is the $2^r \times \ell$ code matrix, generated using an $[\ell, r]$- linear coding scheme, with *binary phase-shift keying* mapping and scaled by $2^{-r/2}$ such that all columns have unit norm.

- Binary Phase-Shift Keying (BPSK): Given a codeword $c \in C$, $c \mapsto \phi \in \mathbb{R}^\ell$ by assigning $1 \to \frac{-1}{\sqrt{2^r}}$ and $0 \to \frac{1}{\sqrt{2^r}}$.

- Define the *dual* of a BCH code as a code of length $\ell' = 2^q - 1 = \ell$, dimension $r' = tq$ and minimum distance at least $2^{q-1} - (t-1)2^{q/2}$.
- **Fact:** Any error correcting code matrix with dual distance > 4 (more than 2 error correcting ability) will satisfy the Johnson-Lindenstrauss Transform (JLT) property.

Lemma

Let $0 < \epsilon, \delta < 1$ and f be some function. It $\Omega \in \mathbb{R}^{n\times\ell}$ satisfies a JLT-(ϵ, δ, d) with $\ell = O(k\log(k/\epsilon)/\epsilon^2.f(\delta))$, then for any orthonormal matrix $V \in \mathbb{R}^{n\times k}, n \geq k$ we have

$$\boldsymbol{Pr}(||V^T\Omega\Omega^T V - I||_2 \leq \epsilon) \geq 1 - \delta.$$

Nice Properties of SCM

- Define the *dual* of a BCH code as a code of length $\ell' = 2^q - 1 = \ell$, dimension $r' = tq$ and minimum distance at least $2^{q-1} - (t-1)2^{q/2}$.

- **Fact:** Any error correcting code matrix with dual distance > 4 (more than 2 error correcting ability) will satisfy the Johnson-Lindenstrauss Transform (JLT) property.

Lemma

Let $0 < \epsilon, \delta < 1$ and f be some function. It $\Omega \in \mathbb{R}^{n \times \ell}$ satisfies a JLT-(ϵ, δ, d) with $\ell = O(k \log(k/\epsilon)/\epsilon^2 . f(\delta))$, then for any orthonormal matrix $V \in \mathbb{R}^{n \times k}$, $n \geq k$ we have

$$\boldsymbol{Pr}(||V^T \Omega \Omega^T V - I||_2 \leq \epsilon) \geq 1 - \delta.$$

- Above lemma shows that, any sampling matrix Ω satisfying JLT and having length $\ell = O(k \log(k/\epsilon)/\epsilon^2)$ satisfies the subspace embedding property.

- Define the *dual* of a BCH code as a code of length $\ell' = 2^q - 1 = \ell$, dimension $r' = tq$ and minimum distance at least $2^{q-1} - (t-1)2^{q/2}$.

- **Fact:** Any error correcting code matrix with dual distance > 4 (more than 2 error correcting ability) will satisfy the Johnson-Lindenstrauss Transform (JLT) property.

Lemma

Let $0 < \epsilon, \delta < 1$ and f be some function. It $\Omega \in \mathbb{R}^{n \times \ell}$ satisfies a JLT-(ϵ, δ, d) with $\ell = O(k \log(k/\epsilon)/\epsilon^2.f(\delta))$, then for any orthonormal matrix $V \in \mathbb{R}^{n \times k}$, $n \geq k$ we have

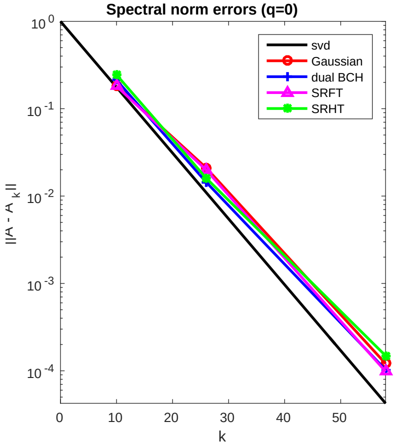
$$\boldsymbol{Pr}(||V^T \Omega \Omega^T V - I||_2 \leq \epsilon) \geq 1 - \delta.$$

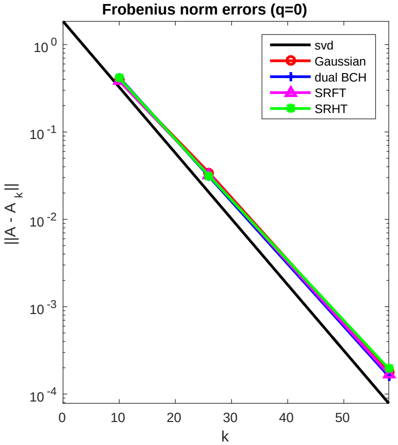
- Above lemma shows that, any sampling matrix Ω satisfying JLT and having length $\ell = O(k \log(k/\epsilon)/\epsilon^2)$ satisfies the subspace embedding property.

- Any SCM Ω with a dual distance > 4 will also satisfy the subspace embedding property. This shows SCM matrices can preserve the geometry of the top k-singular vectors of input matrix A.

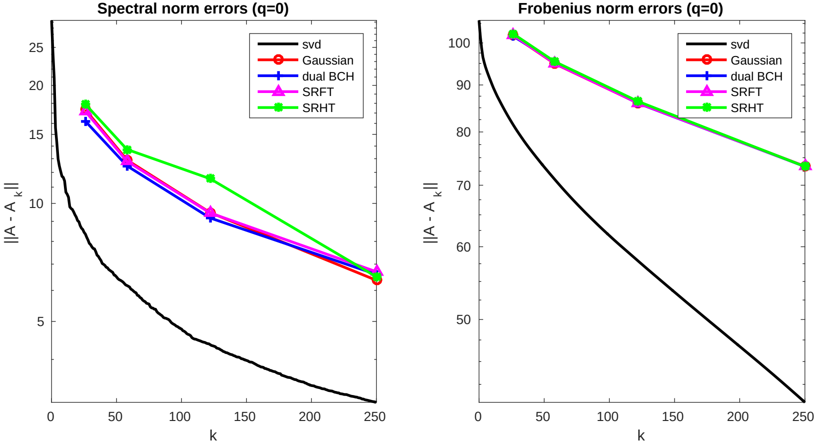
Sampling for the RSVD

- We implemented dual BCH codes in MATLAB using tools from the Communications Systems toolbox.
- We only need an encoder; decoders are more complicated.
- `LOCAL_fast_decay`: 100×140.





- `Kohonen.mat`: 4470 $\times$ 4470, an adjacency matrix for directed graph.



The Hadamard Transform

The (naturally ordered) Hadamard transform is defined recursively for sizes $d = 2^L$:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

$$H_d = \begin{bmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & -H_{d/2} \end{bmatrix}.$$

Using this recursion, we have a fast algorithm (similar to FFT and Haar wavelet).
Runtime is $\mathcal{O}(d \log d)$.

Theorem

Every column of the $2^r \times \ell$ code matrix Φ (after BPSK mapping) is equal to some column of the $2^r \times 2^r$ Hadamard matrix.

So, we can apply the SCM in $\mathcal{O}(d \log d)$ time.

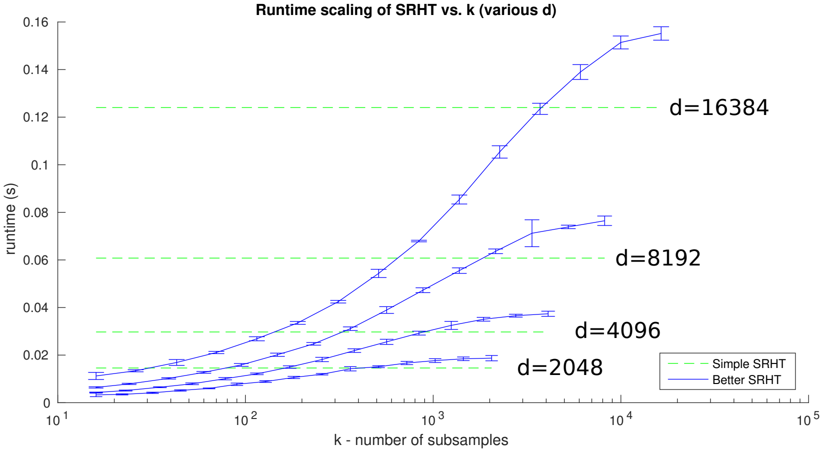
Like the SRFT, we can define an SRHT as $P_k H_d x$. Unfortunately, no one has actually implemented and published an SRHT!
Naïve subsampling costs $\mathcal{O}(d \log d)$, independent of the number of subsamples k.

By splitting $P_k H_d x$, we find an efficient algorithm for the SRHT [**?**]:

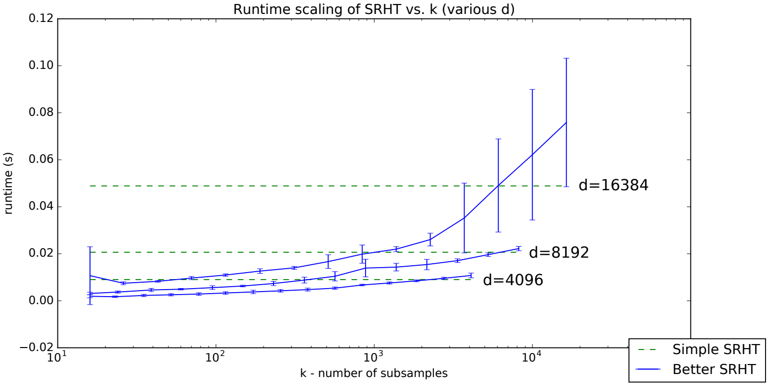
$$P_k H_d x = \begin{bmatrix} P_{k_1} & P_{k_2} \end{bmatrix} \begin{bmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & -H_{d/2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = P_{k_1} H_{d/2}(x_1 + x_2) + P_{k_2} H_{d/2}(x_1 - x_2).$$

It can be shown that this runs in $\mathcal{O}(d \log k)$ time, which is an improvement over the naïve $\mathcal{O}(d \log d)$.

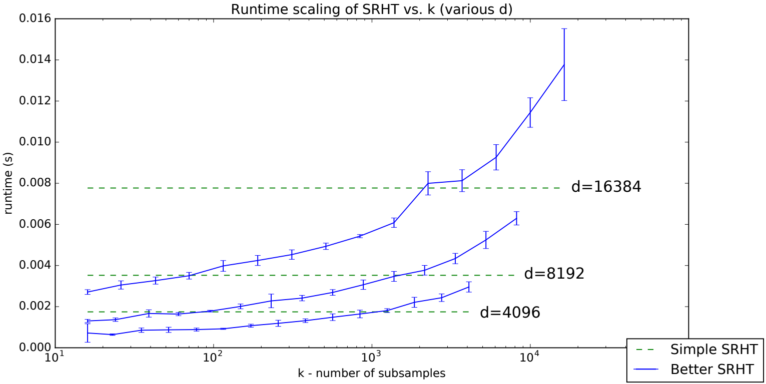
Reference MATLAB implementations:



Reference Julia implementations:



Optimized C+SSE2 implementations:



Bibliography I

Our code is online at

`https://github.com/jamesfolberth/fast_methods_big_data_project.`

[1] N. Ailon and E. Liberty.
Fast dimension reduction using rademacher series on dual bch codes.
Discrete & Computational Geometry, 42(4):615–630, 2009.

[2] J. I. Hall.
Notes on coding theory.
FreeTechBooks. com, 2003.

[3] S. Ubaru, A. Mazumdar, and Y. Saad.
Low rank approximation using error correcting coding matrices.
In Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pages 702–710, 2015.