CUR Matrix Factorizations

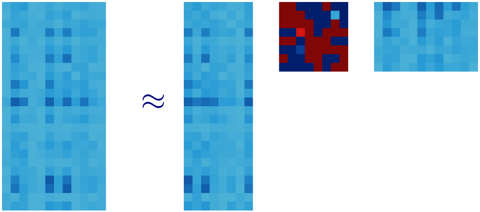
Algorithms, Analysis, Applications

Mark Embree · Virginia Tech

embree@vt.edu

October 2017

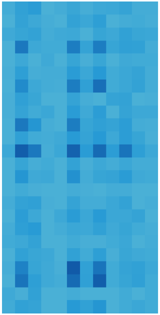
Based on:
D. C. Sorensen and M. Embree. "A DEIM Induced CUR Factorization."
SIAM J. Sci. Comput. 38 (2016) A1454–A1482.
http://epubs.siam.org/doi/pdf/10.1137/140978430



Low rank data decompositions

Begin with data stored in a matrix with (approximate) low-rank structure.
We seek to factor **A** as a product of matrices that reveals this structure.

A



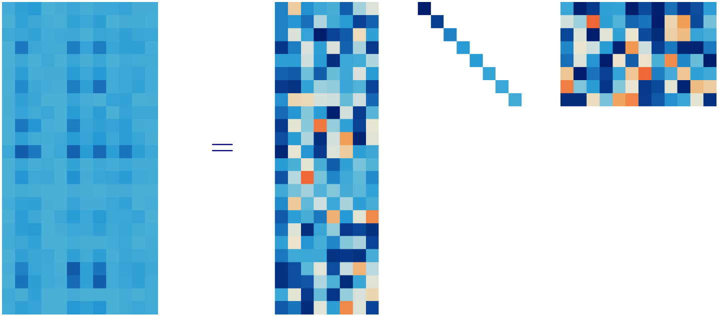
Low rank data decompositions

Begin with data stored in a matrix with (approximate) low-rank structure.
We seek to factor $\mathbf{A}$ as a product of matrices that reveals this structure.

The classic, *optimal* approach is the *singular value decomposition*,
with $\mathbf{V}$ and $\mathbf{W}$ having orthonormal columns, and $\boldsymbol{\Sigma}$ diagonal.

However, the orthonormal singular vectors aren't so representative of the data.

$$\mathbf{A} \quad = \quad \mathbf{V} \quad \boldsymbol{\Sigma} \quad \mathbf{W}^*$$



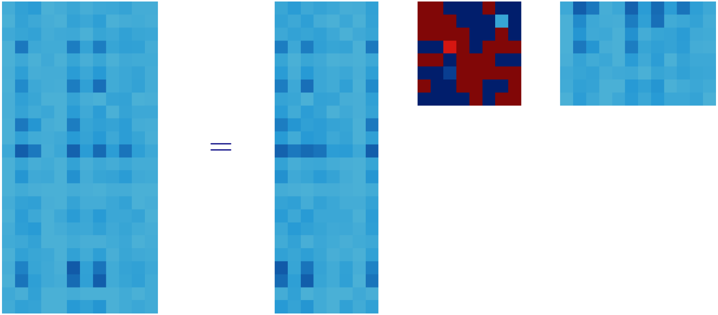
Low rank data decompositions

Begin with data stored in a matrix with (approximate) low-rank structure.
We seek to factor **A** as a product of matrices that reveals this structure.

This talk concerns *suboptimal* approaches called *interpolatory factorizations*.
The best known example is the CUR decomposition.

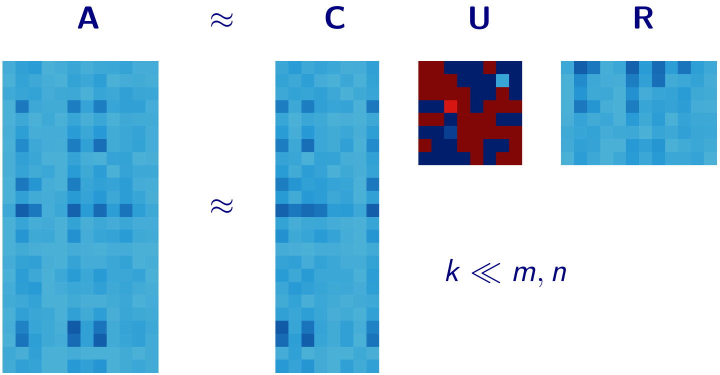
C and **R** are taken from the columns and rows of **A**, so they are representative.

$$\mathbf{A} \quad = \quad \mathbf{C} \quad \mathbf{U} \quad \mathbf{R}$$



Interpolatory approximations

We generally only seek approximations. (Noisy data makes **A** full rank.)



$\mathbf{A} \in \mathbb{R}^{m \times n}$

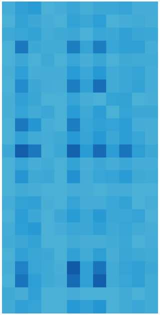
$\mathbf{C} \in \mathbb{R}^{m \times k}$ is a *subset of the columns of* **A**
$\mathbf{U} \in \mathbb{R}^{k \times k}$ optimizes the approximation
$\mathbf{R} \in \mathbb{R}^{k \times n}$ is a *subset of the rows of* **A**

U can be ill-conditioned – but we often only care about columns *or* rows.

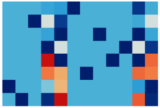
Interpolatory approximations

Consider flexible *interpolatory approximations* [Martinsson, Rokhlin, Tygert 2011].

$$\mathbf{A} \approx \mathbf{C} \quad \mathbf{X}$$



$\approx$



$k \ll m, n$

$\mathbf{A} \in \mathbb{R}^{m \times n}$

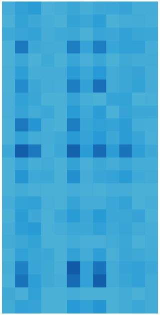
$\mathbf{C} \in \mathbb{R}^{m \times k}$ is a *subset of the columns of* $\mathbf{A}$
$\mathbf{X} \in \mathbb{R}^{k \times n}$ optimizes the approximation

Only extract columns; more flexible structure can give better-conditioned $\mathbf{X}$*.*

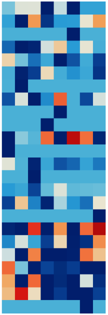
Interpolatory approximations

Consider flexible *interpolatory approximations* [Martinsson, Rokhlin, Tygert 2011].

$$\mathbf{A} \approx \mathbf{X} \quad \mathbf{R}$$



$\approx$



$k \ll m, n$

$\mathbf{A} \in \mathbb{R}^{m \times n}$

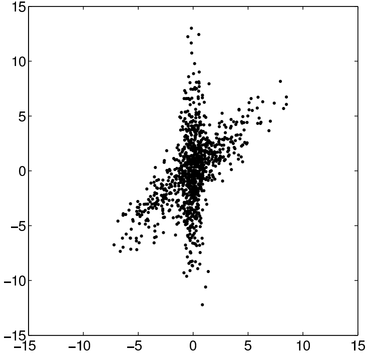
$\mathbf{X} \in \mathbb{R}^{m \times k}$ optimizes the approximation
$\mathbf{R} \in \mathbb{R}^{k \times n}$ is a *subset of the rows of* $\mathbf{A}$

Only extract rows; more flexible structure can give better-conditioned $\mathbf{X}$*.*

A simple motivation for CUR factorizations

A simple illustrative example from Mahoney and Drineas [2009].

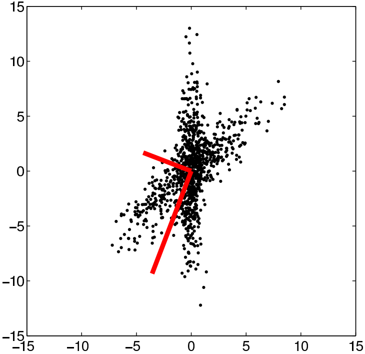
Suppose the rows of $\mathbf{A} \in \mathbb{R}^{m \times 2}$ are from one of two multivariate normal distributions of similar magnitude. Can we detect the two main axes?



A simple motivation for CUR factorizations

A simple illustrative example from Mahoney and Drineas [2009].

Suppose the rows of $\mathbf{A} \in \mathbb{R}^{m \times 2}$ are from one of two multivariate normal distributions of similar magnitude. Can we detect the two main axes?

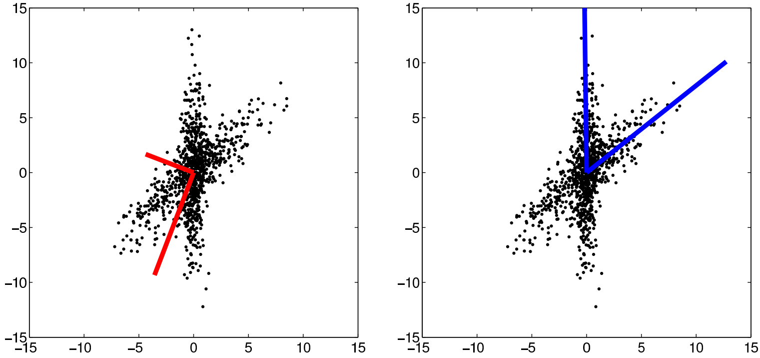


The singular vectors miss both primary directions.

A simple motivation for CUR factorizations

A simple illustrative example from Mahoney and Drineas [2009].

Suppose the rows of $\mathbf{A} \in \mathbb{R}^{m \times 2}$ are from one of two multivariate normal distributions of similar magnitude. Can we detect the two main axes?



The singular vectors miss both primary directions.

The two rows of $\mathbf{R}$ capture representative data.

Example: Supreme Court voting patterns

Use of CUR to identify distinctive voting patterns on the Rehnquist court; cf. [Sirovich, *PNAS*, 2003].

- $\mathbf{A} \in \mathbb{R}^{493 \times 9}$: data for 493 cases, 9 justices (from Keith Poole, U. Georgia)
- (j, k) entry $= 1$ if justice k voted with the majority on case j
- (j, k) entry $= 0$ if justice k voted in the dissent on case j

Example: Supreme Court voting patterns

Use of CUR to identify distinctive voting patterns on the Rehnquist court; cf. [Sirovich, *PNAS*, 2003].

- $\mathbf{A} \in \mathbb{R}^{493\times 9}$: data for 493 cases, 9 justices (from Keith Poole, U. Georgia)
- (j,k) entry $= 1$ if justice k voted with the majority on case j
- (j,k) entry $= 0$ if justice k voted in the dissent on case j

First six rows selected by the DEIM-CUR factorization:

	Rehnquist	Stevens	O'Connor	Scalia	Kennedy	Souter	Thomas	Ginsburg	Breyer
1	+	+	+	+	+	+	+	+	+
2	+	o	+	+	+	o	+	o	o
3	+	o	+	o	+	+	o	+	+
4	o	+	+	o	+	o	+	o	+
5	+	o	+	+	o	o	+	o	+
6	+	+	o	o	+	o	o	+	+

CUR Factorizations

CUR approximations can be computed using various different strategies.

- Column pivoted QR factorizations [Stewart 1999],
  [Voronin and Martinsson 2015]
  cf. Rank Revealing QR factorizations [Gu, Eisenstat 1996]

- Volume optimization [Goreinov, Tyrtyshnikov, Zamarashkin 1997], . . . ,
  [Goreinov, Oseledets, Savostyanov, Tyrtyshnikov, Zamarashkin 2010],
  [Thurau, Kersting, Bauckhage 2012]

- Uniform sampling of columns e.g., [Chiu, Demanet 2012]

- Leverage scores (norms of rows of singular vector matrices)
  [Drineas, Mahoney, Muthukrishnan 2008], [Mahoney, Drineas 2009], . . . ,
  [Boutsidis, Woodruff 2014]

- Empirical Interpolation approaches
  [Sorensen & E.], Q-DEIM method of [Drmač, Gugercin 2015]

The last two classes of methods use (approximate) singular vectors.

CUR Factorization: Goals

Let $\mathbf{A} \approx \mathbf{CUR}$ be an approximate rank-k factorization of $\mathbf{A}$; $r = \mathrm{rank}(\mathbf{A})$.

- Since the SVD is optimal, we must have

$$\begin{aligned} \|\mathbf{A} - \mathbf{CUR}\|_2 &\geq \sigma_{k+1} \\ \|\mathbf{A} - \mathbf{CUR}\|_F &\geq \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}. \end{aligned}$$

CUR Factorization: Goals

Let $\mathbf{A} \approx \mathbf{CUR}$ be an approximate rank-k factorization of $\mathbf{A}$; $r = \mathrm{rank}(\mathbf{A})$.

- Since the SVD is optimal, we must have

$$\|\mathbf{A} - \mathbf{CUR}\|_2 \geq \sigma_{k+1}$$

$$\|\mathbf{A} - \mathbf{CUR}\|_F \geq \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

- We seek algorithms that compute $\mathbf{C} \in \mathbb{R}^{m\times k}$, $\mathbf{R} \in \mathbb{R}^{k\times n}$ for which

$$\|\mathbf{A} - \mathbf{CUR}\|_2 \leq C_2\ \sigma_{k+1}$$

$$\|\mathbf{A} - \mathbf{CUR}\|_F \leq C_F\ \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

for some 'modest' constants C_2 or C_F.

CUR Factorization: Goals

Let $\mathbf{A} \approx \mathbf{CUR}$ be an approximate rank-k factorization of $\mathbf{A}$; $r = \text{rank}(\mathbf{A})$.

- Since the SVD is optimal, we must have

$$\begin{aligned} \|\mathbf{A} - \mathbf{CUR}\|_2 &\geq \sigma_{k+1} \\ \|\mathbf{A} - \mathbf{CUR}\|_F &\geq \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}. \end{aligned}$$

- We seek algorithms that compute $\mathbf{C} \in \mathbb{R}^{m\times k}$, $\mathbf{R} \in \mathbb{R}^{k\times n}$ for which

$$\begin{aligned} \|\mathbf{A} - \mathbf{CUR}\|_2 &\leq C_2\ \sigma_{k+1} \\ \|\mathbf{A} - \mathbf{CUR}\|_F &\leq C_F\ \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}. \end{aligned}$$

  for some 'modest' constants C_2 or C_F.

- In contrast, the theoretical computer science community often seeks, for any specified $\varepsilon \in (0,1)$, matrices $\mathbf{C} \in \mathbb{R}^{m\times p}$, $\mathbf{R} \in \mathbb{R}^{q\times m}$ such that

$$\|\mathbf{A} - \mathbf{CUR}\|_F^2 \leq (1+\varepsilon)\Big(\sigma_{k+1}^2 + \cdots + \sigma_r^2\Big);$$

  for $p, q = O(k/\varepsilon)$, $\text{rank}(\mathbf{U}) = k$; see, e.g., [Boutsidis, Woodruff 2014].

Outline

Fundamental questions:

- Which columns of $\mathbf{A}$ should form $\mathbf{C}$? Which rows should form $\mathbf{R}$?
- Given $\mathbf{C}$ and $\mathbf{R}$, how should we best construct $\mathbf{U}$?

Plan for this talk:

- DEIM as a method for fast basis selection
- DEIM-induced CUR factorization
- Analysis of CUR factorizations via interpolatory projectors
- Examples

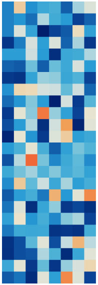
CUR Row Selection based on Singular Vectors

CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\boldsymbol{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



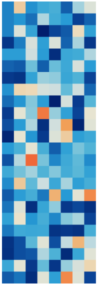
$\mathbf{V}$

CUR based on Leverage Scores

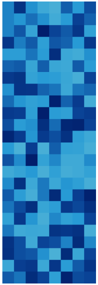
Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\mathbf{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



$\mathbf{V}$



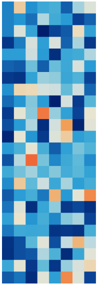
$|\mathbf{V}(j,k)|$

CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\boldsymbol{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



$\mathbf{V}$

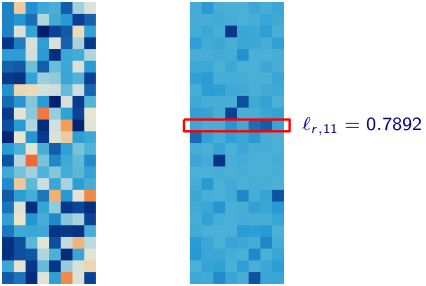
$|\mathbf{V}(j,k)|^2$

CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V\Sigma W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



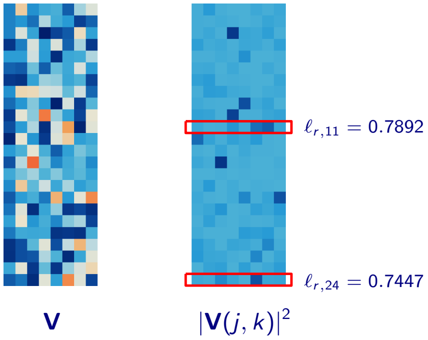
$\mathbf{V}$ $\quad$ $|\mathbf{V}(j,k)|^2$

CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\mathbf{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$

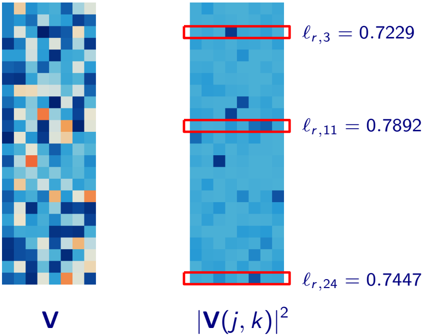


CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V\Sigma W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$

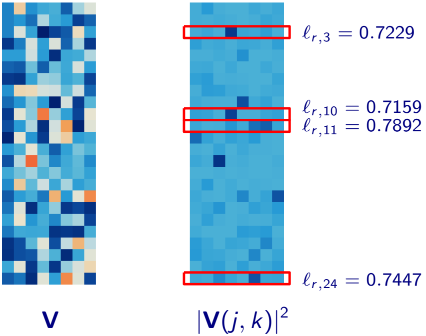


CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\mathbf{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$

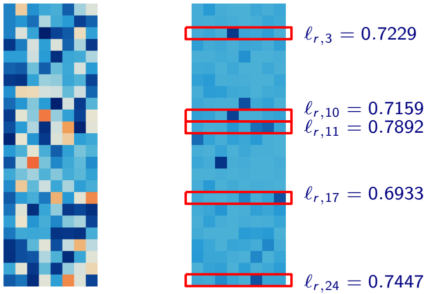


CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V}\mathbf{\Sigma}\mathbf{W}^*$, $\mathbf{V} \in \mathbb{R}^{m \times r}$, $\mathbf{W} \in \mathbb{R}^{n \times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



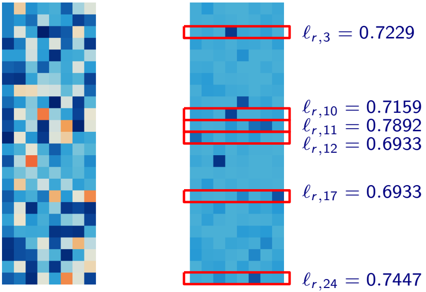
$\mathbf{V}$ $\quad$ $|\mathbf{V}(j,k)|^2$

CUR based on Leverage Scores

Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V\Sigma W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



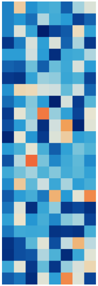
$\mathbf{V}$ $\qquad$ $|\mathbf{V}(j,k)|^2$

CUR based on Leverage Scores

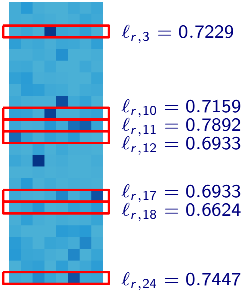
Leverage Scores are a popular technique for computing the CUR factorization, based on identifying the key elements of the singular vectors; see, e.g., [Mahoney, Drineas 2009].

- Suppose we have $\mathbf{A} = \mathbf{V\Sigma W}^*$, $\mathbf{V} \in \mathbb{R}^{m\times r}$, $\mathbf{W} \in \mathbb{R}^{n\times r}$.
- To rank the importance of the *rows*, take the 2-norm of each *row* of $\mathbf{V}$:

$$\text{row leverage score} = \ell_{r,j} = \|\mathbf{V}(j,:)\|_2.$$



$\mathbf{V}$



$|\mathbf{V}(j,k)|^2$

CUR based on Leverage Scores

- To get $\mathbf{R} \in \mathbb{R}^{m \times k}$, extract the rows of $\mathbf{A}$ with k highest leverage scores.
- Or, use the fact that

$$\sum_{j=1}^{m} \ell_{r,j}^2 = n$$

to get a probability distribution $\{\ell_{r,j}^2/n\}$ for *random row selection*.

CUR based on Leverage Scores

- To get $\mathbf{R} \in \mathbb{R}^{m \times k}$, extract the rows of $\mathbf{A}$ with k highest leverage scores.
- Or, use the fact that

$$\sum_{j=1}^{m} \ell_{r,j}^2 = n$$

  to get a probability distribution $\{\ell_{r,j}^2/n\}$ for *random row selection*.

- To rank the importance of the *columns*, take the 2-norm of each *row* of $\mathbf{W}$:

$$\text{row leverage score} = \ell_{c,j} = \|\mathbf{W}(j,:)\|.$$

- Construct $\mathbf{C}$ as the columns that have the highest leverage score (or use random selection).

CUR based on Leverage Scores

- To get $\mathbf{R} \in \mathbb{R}^{m \times k}$, extract the rows of $\mathbf{A}$ with k highest leverage scores.
- Or, use the fact that
$$\sum_{j=1}^{m} \ell_{r,j}^2 = n$$
to get a probability distribution $\{\ell_{r,j}^2/n\}$ for *random row selection*.

- To rank the importance of the *columns*, take the 2-norm of each *row* of $\mathbf{W}$:
$$\text{row leverage score} = \ell_{c,j} = \|\mathbf{W}(j,:)\|.$$
- Construct $\mathbf{C}$ as the columns that have the highest leverage score (or use random selection).

- Leverage scores can be highly influenced by latter columns of $\mathbf{V}$ and $\mathbf{W}$ that correspond to the *smaller* singular values.
- One can compute leverage scores using the leading columns of $\mathbf{V}$ and $\mathbf{W}$.
- A perturbation theory has been developed by [Ipsen, Wentworth 2014].

DEIM for Row and Column Selection

We shall pick columns and rows of **A** to form **C** and **R** using a variant of the *Discrete Empirical Interpolation Method* (DEIM) method.

- DEIM was proposed by [Chaturantabut, Sorensen 2010] for model order reduction of nonlinear dynamical systems.
- DEIM is based upon the *Empirical Interpolation Method* of [Barrault, Maday, Nguyen, Patera 2004], which was presented in the context of finite element methods.
- The Q-DEIM variant algorithm of [Drmač, Gugercin 2015] can be readily adapted to give CUR factorizations; see [Saibaba 2015] for an extension of these ideas to tensors.

Key Tool: Interpolatory Projectors

Let $\mathbf{V} \in \mathbb{R}^{m \times k}$ have orthonormal columns,
and let $p_1, \ldots, p_k \in \{1, \ldots, m\}$ denote a set of k distinct row indices.

Key Tool: Interpolatory Projectors

Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ have orthonormal columns,
and let $p_1, \ldots, p_k \in \{1, \ldots, m\}$ denote a set of k distinct row indices.

We are accustomed to working with the *orthogonal projector*

$$\mathbf{\Pi} = \mathbf{V}(\mathbf{V}^\mathsf{T}\mathbf{V})^{-1}\mathbf{V}^\mathsf{T} = \mathbf{V}\mathbf{V}^\mathsf{T}.$$

here we work with the *interpolatory projector*

$$\mathcal{P} = \mathbf{V}(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\mathbf{E}^\mathsf{T},$$

where $\mathbf{E} = \mathbf{I}(\,:\,,\mathbf{p}) = [\mathbf{e}_{p_1}\ \mathbf{e}_{p_2}\ \cdots\ \mathbf{e}_{p_k}] \in \mathbb{R}^{m\times k}$.

Key Tool: Interpolatory Projectors

Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ have orthonormal columns,
and let $p_1, \ldots, p_k \in \{1, \ldots, m\}$ denote a set of k distinct row indices.

We are accustomed to working with the *orthogonal projector*

$$\mathbf{\Pi} = \mathbf{V}(\mathbf{V}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{V}^{\mathsf{T}} = \mathbf{V}\mathbf{V}^{\mathsf{T}}.$$

here we work with the *interpolatory projector*

$$\mathcal{P} = \mathbf{V}(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{E}^{\mathsf{T}},$$

where $\mathbf{E} = \mathbf{I}(\,:\,,\mathbf{p}) = [\mathbf{e}_{p_1}\ \mathbf{e}_{p_2}\ \cdots\ \mathbf{e}_{p_k}] \in \mathbb{R}^{m\times k}$.

For example, if $p_1 = 6$, $p_2 = 3$, and $p_3 = 1$, we have

$$\mathbf{E}^{\mathsf{T}}\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} x_6 \\ x_3 \\ x_1 \end{bmatrix}$$

Key Tool: Interpolatory Projectors

Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ have orthonormal columns,
and let $p_1, \dots, p_k \in \{1, \dots, m\}$ denote a set of k distinct row indices.

We are accustomed to working with the *orthogonal projector*

$$\mathbf{\Pi} = \mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T = \mathbf{V}\mathbf{V}^T.$$

here we work with the *interpolatory projector*

$$\mathcal{P} = \mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\mathbf{E}^T,$$

where $\mathbf{E} = \mathbf{I}(:,\mathbf{p}) = [\mathbf{e}_{p_1}\ \mathbf{e}_{p_2}\ \cdots\ \mathbf{e}_{p_k}] \in \mathbb{R}^{m\times k}$.

We call $\mathcal{P}$ *interpolatory* because $\mathcal{P}\mathbf{x}$ matches $\mathbf{x}$ (for any $\mathbf{x}$) in its $\mathbf{p}$ entries:

$$(\mathcal{P}\mathbf{x})(\mathbf{p}) = \mathbf{x}(\mathbf{p}),$$

i.e.,

$$(\mathcal{P}\mathbf{x})(\mathbf{p}) = \mathbf{E}^T\mathcal{P}\mathbf{x} = \mathbf{E}^T\mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\mathbf{E}^T\mathbf{x} = \mathbf{E}^T\mathbf{x} = \mathbf{x}(\mathbf{p}).$$

$$\mathcal{P}\begin{bmatrix} x_1 \\ \times \\ x_3 \\ \times \\ \times \\ x_6 \\ \times \end{bmatrix} = \begin{bmatrix} x_1 \\ \# \\ x_3 \\ \# \\ \# \\ x_6 \\ \# \end{bmatrix}$$

Key Tool: Interpolatory Projectors

The *orthogonal projector*

$$\mathbf{\Pi} = \mathbf{V}(\mathbf{V}^\mathsf{T}\mathbf{V})^{-1}\mathbf{V}^\mathsf{T} = \mathbf{V}\mathbf{V}^\mathsf{T}$$

and the (oblique) *interpolatory projector*

$$\mathcal{P} = \mathbf{V}(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\mathbf{E}^\mathsf{T}$$

are both projectors

$$\mathbf{\Pi}^2 = \mathbf{\Pi} \qquad \mathcal{P}^2 = \mathcal{P}$$

onto the same subspace

$$\mathrm{Ran}(\mathbf{\Pi}) = \mathrm{Ran}(\mathcal{P}) = \mathrm{Ran}(\mathbf{V}) = \mathrm{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}.$$

We will build up $\mathcal{P}$ by finding interpolation indices $p_1, \ldots, p_k$ one at a time.

Key Tool: Interpolatory Projectors

The *orthogonal projector*

$$\boldsymbol{\Pi} = \mathbf{V}(\mathbf{V}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{V}^{\mathsf{T}} = \mathbf{V}\mathbf{V}^{\mathsf{T}}$$

and the (oblique) *interpolatory projector*

$$\mathcal{P} = \mathbf{V}(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{E}^{\mathsf{T}}$$

are both projectors

$$\boldsymbol{\Pi}^2 = \boldsymbol{\Pi} \qquad \mathcal{P}^2 = \mathcal{P}$$

onto the same subspace

$$\mathrm{Ran}(\boldsymbol{\Pi}) = \mathrm{Ran}(\mathcal{P}) = \mathrm{Ran}(\mathbf{V}) = \mathrm{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}.$$

We will build up $\mathcal{P}$ by finding interpolation indices $p_1, \ldots, p_k$ one at a time.

In the following, let

$$\begin{aligned} \mathbf{V}_j &= \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_j \end{bmatrix} \in \mathbb{R}^{m\times j} \\ \mathbf{E}_j &= \mathbf{I}(\,:, [p_1, \ldots, p_j]) \in \mathbb{R}^{m\times j}. \end{aligned}$$

Find indices via a (non-orthogonal) Gram–Schmidt-like process on $\mathbf{v}_1, \ldots, \mathbf{v}_k$.

Discrete Empirical Interpolation Method (DEIM)

Goal: Find indices $p_1, \ldots, p_k$ identifying the most prominent rows in $\mathbf{V}_k$.

Step 1: Set p_1 to the largest entry in the dominant singular vector:

$$p_1 = \arg \max_{1 \leq j \leq m} |(\mathbf{v}_1)_j|$$

$$\mathbf{v}_1 = \begin{array}{c@{}l} \left[\begin{array}{c} \times \\ \times \\ \times \\ \times \\ \times \\ \bigstar \\ \times \end{array}\right] & \begin{array}{l} \\ \\ \\ \\ \\ p_1 \\ \\ \end{array} \end{array}$$

Discrete Empirical Interpolation Method (DEIM)

Goal: Find indices $p_1, \ldots, p_k$ identifying the most prominent rows in $\mathbf{V}_k$.

Step 2: Find p_2 by removing the $\mathbf{v}_1$ component from $\mathbf{v}_2$.

Step 2a: Construct the interpolatory projector for p_1:

$$\mathcal{P}_1 = \mathbf{v}_1(\mathbf{E}_1^T\mathbf{v}_1)^{-1}\mathbf{E}_1^T.$$

Step 2b: Project $\mathbf{v}_1$ against $\mathbf{v}_2$ to zero out p_1 entry, and compute the residual:

$$\mathbf{r}_2 = \mathbf{v}_2 - \mathcal{P}_1\mathbf{v}_2.$$

Step 2c: Identify the largest entry in the residual:

$$p_2 = \arg\max_{1\le j\le m} |(\mathbf{r}_2)_j|.$$

$$\mathbf{r}_2 = \left[\begin{array}{c} \times \\ \times \\ \bigstar \\ \times \\ \times \\ {\color{red}0} \\ \times \end{array}\right] \begin{array}{l} \\ \\ p_2 \\ \\ \\ \\ \\ \end{array}$$

Discrete Empirical Interpolation Method (DEIM)

Goal: Find indices $p_1, \ldots, p_k$ identifying the most prominent rows in $\mathbf{V}_k$.

Step 3: Find p_3 by removing the $\mathbf{v}_1$ and $\mathbf{v}_2$ components from $\mathbf{v}_3$.

Step 3a: Construct the interpolatory projector for p_1 and p_2:

$$\mathcal{P}_2 = \mathbf{V}_2(\mathbf{E}_2^T\mathbf{V}_2)^{-1}\mathbf{E}_2^T.$$

Step 3b: Project $\mathbf{v}_1$ and $\mathbf{v}_2$ against $\mathbf{v}_3$ to zero out the p_1 and p_2 entries, and compute the residual:

$$\mathbf{r}_3 = \mathbf{v}_3 - \mathcal{P}_2\mathbf{v}_3.$$

Step 3c: Identify the largest entry in the residual:

$$p_3 = \arg\max_{1\le j\le m} |(\mathbf{r}_3)_j|.$$

$$\mathbf{r}_3 = \begin{array}{cl} \left[\begin{array}{c} \bigstar \\ \times \\ 0 \\ \times \\ \times \\ 0 \\ \times \end{array}\right] & \begin{array}{l} p_3 \\ \\ \\ \\ \\ \\ \\ \end{array} \end{array}$$

Discrete Empirical Interpolation Method (DEIM)

The index selection process is very simple.

DEIM Row Selection Process

Input: $\mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathbb{R}^m$, with $\mathbf{V}_j = [\mathbf{v}_1\ \mathbf{v}_2\ \cdots\ \mathbf{v}_j]$

Output: $\mathbf{p} \in \mathbb{R}^k$ (unique indices in $\{1, \ldots, m\}$)

$[\sim, p_1] = \max|\mathbf{v}_1|$

$\mathbf{p} = [p_1]$

for $j = 2, \ldots, k$

$\quad \mathbf{r} = \mathbf{v}_j - \mathbf{V}_{j-1}\Big(\mathbf{V}_{j-1}(\mathbf{p}, :)^{-1}\mathbf{v}_j(\mathbf{p})\Big)$

$\quad [\sim, p_j] = \max|\mathbf{r}|$

$\quad \mathbf{p} = [\mathbf{p}; p_j]$

end

Discrete Empirical Interpolation Method (DEIM)

- DEIM closely resembles Gaussian elimination with partial pivoting, and this informs the worst-case error analysis described later.
- DEIM algorithm can be stopped at any k, e.g., as soon as adequate approximation is found.
- Q-DEIM variant applies column-pivoted QR factorization to $\mathbf{V}_k^\mathsf{T}$, using the pivot columns as the interpolation indices. If k is fixed ahead of time, this gives a *basis-independent* way to pick the k pivots [Drmač, Gugercin 2015].

The DEIM-CUR Approximate Factorization

CUR Factorization using DEIM

To compute the CUR-DEIM factorization:

- Compute/approximate the dominant left and right singular vectors, $\mathbf{V} \in \mathbb{R}^{m \times k}$, $\mathbf{W} \in \mathbb{R}^{n \times k}$.
- Select row indices $\mathbf{p}$ by applying DEIM to $\mathbf{V}$.
- Select column indices $\mathbf{q}$ by applying DEIM to $\mathbf{W}$.
- Extract rows, $\mathbf{R} = \mathbf{E}^{\mathsf{T}}\mathbf{A} = \mathbf{A}(\mathbf{p}, :)$.
- Extract columns, $\mathbf{C} = \mathbf{A}\mathbf{F} = \mathbf{A}(:, \mathbf{q})$.

Options for constructing U

Once $\mathbf{C}$ and $\mathbf{R}$ have been constructed, two notable choices are available for $\mathbf{U}$ (presuming one needs the explicit $\mathbf{A} \approx \mathbf{CUR}$ factorization).

- $\mathbf{U} = (\mathbf{E}^\top \mathbf{A} \mathbf{F})^{-1} = (\mathbf{A}(\mathbf{p}, \mathbf{q}))^{-1}$
  This choice is efficient to compute, and it perfectly recovers entries of $\mathbf{A}$:
  $$(\mathbf{CUR})(p, q) = \mathbf{A}(p, q)$$
  for all $p \in \{p_1, \ldots, p_k\}$ and $q \in \{q_1, \ldots, q_k\}$.

- $\mathbf{U} = \mathbf{C}^+ \mathbf{A} \mathbf{R}^+$
  This choice is optimal in the Frobenius norm [Stewart, 1999];
  See also [Mahoney, Drineas, 2009].
  $\mathbf{CUR} = (\mathbf{CC}^+)\mathbf{A}(\mathbf{R}^+\mathbf{R})$, where $\mathbf{CC}^+$ and $\mathbf{R}^+\mathbf{R}$ are orthogonal projectors.
  $\mathbf{C}$ and $\mathbf{R}$ need not have the same number of columns and rows.

Our analysis shall use the latter choice of $\mathbf{U}$. However, we emphasize that the motivating application may not need $\mathbf{U} \in \mathbb{C}^{k \times k}$ explicitly.

Options for computing the SVD

Problem size dictates how to compute/approximate the SVD that feeds DEIM.

- For modest m or n, use the economy SVD: `[V,S,W] = svd(A,'econ')`.

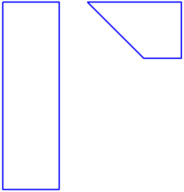
- Krylov SVD routines compute the largest k singular vectors (`svds`).
  These algorithms access $\mathbf{A}$ and $\mathbf{A}^T$ through matrix-vector products.
  Need to access $\mathbf{A}$ often, but need minimal intermediate storage.

- Randomized range-finding techniques can find $\mathbf{V}$ with high probability [Halko, Martinsson, Tropp 2011]. These algorithms also access $\mathbf{A}$ and $\mathbf{A}^T$ through matrix-vector products.
  Like Krylov methods: access $\mathbf{A}$ often, need minimal intermediate storage.

- Incremental QR factorization approximates the SVD in *one pass*.
  Given the economy QR factorization $\mathbf{A} = \widehat{\mathbf{Q}}\widehat{\mathbf{R}}$ for $\widehat{\mathbf{Q}} \in \mathbb{R}^{m\times k}$, $\widehat{\mathbf{R}} \in \mathbb{R}^{k\times k}$, compute the SVD $\widehat{\mathbf{R}} = \widehat{\mathbf{V}}\mathbf{\Sigma}\mathbf{W}^*$. Then $\mathbf{A} = (\widehat{\mathbf{Q}}\widehat{\mathbf{V}})\mathbf{\Sigma}\mathbf{W}^*$ is an SVD of $\mathbf{A}$ cf. [Stewart 1999], [Baker, Gallivan, Van Dooren, 2011].
  Intermediate storage depends on the rank and sparsity of $\mathbf{A}$.

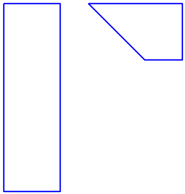
Incremental One-Pass QR Factorization

$\mathbf{A}(:, 1:k) =$

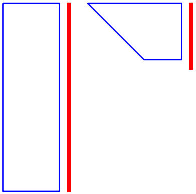


Partial **QR** factorization

Incremental One-Pass QR Factorization

$\mathbf{A}(:, 1:k) =$ 

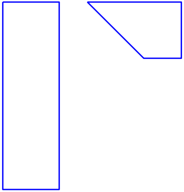
Partial **QR** factorization

$\mathbf{A}(:, 1{:}k{+}1) =$ 

Extend with Gram–Schmidt

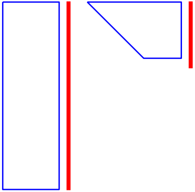
Incremental One-Pass QR Factorization

$\mathbf{A}(:, 1:k) =$

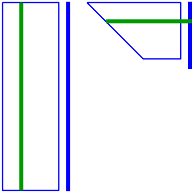


Partial **QR** factorization

$\mathbf{A}(:, 1:k+1) =$



Extend with Gram–Schmidt

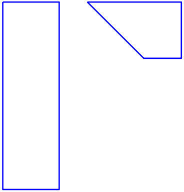


Find $\mathbf{q}_j$ with

$$\|\mathbf{R}(j,:)\|^2 < \varepsilon^2 \left(\|\mathbf{R}\|_F^2 - \|\mathbf{R}(j,:)\|^2\right)$$

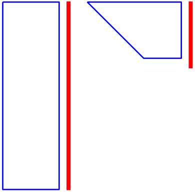
Incremental One-Pass QR Factorization

$\mathbf{A}(:, 1:k) =$

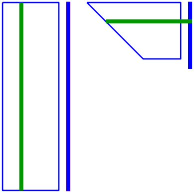


Partial **QR** factorization

$\mathbf{A}(:, 1\!:\!k\!+\!1) =$

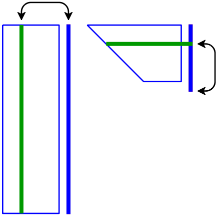


Extend with Gram–Schmidt



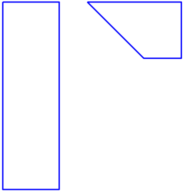
Find $\mathbf{q}_j$ with

$\|\mathbf{R}(j,:)\|^2 < \varepsilon^2 \left(\|\mathbf{R}\|_F^2 - \|\mathbf{R}(j,:)\|^2\right)$

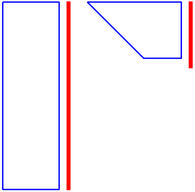


Replace $\mathbf{q}_j$, $\mathbf{R}(j,:)$

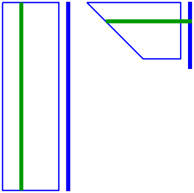
Incremental One-Pass QR Factorization

$\mathbf{A}(:, 1:k) =$ 

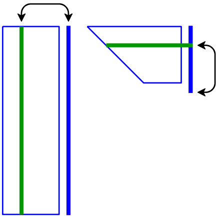
Partial **QR** factorization

$\mathbf{A}(:, 1\!:\!k\!+\!1) =$ 

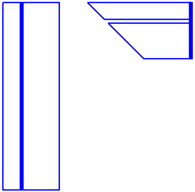
Extend with Gram–Schmidt



Find $\mathbf{q}_j$ with
$\|\mathbf{R}(j,:)\|^2 < \varepsilon^2 \left(\|\mathbf{R}\|_F^2 - \|\mathbf{R}(j,:)\|^2\right)$



Replace $\mathbf{q}_j$, $\mathbf{R}(j,:)$



Truncate last col of **Q**
and last row of **R**

Incremental One-Pass QR Factorization: Analysis

How does badly does this simple truncation strategy compromise the accuracy of the factorization?

Let $\mathbf{A}_k = \mathbf{A}(:, 1:k)$ denote the first k columns of $\mathbf{A}$.

Theorem. Perform k steps of the incremental QR algorithm to get $\mathbf{A}_k \approx \mathbf{Q}_k\mathbf{R}_k$ using d_k deletions governed by the tolerance ε:

$$\mathbf{A}_k \in \mathbb{R}^{n\times k}, \qquad \mathbf{Q}_k \in \mathbb{R}^{n\times(k-d_k)}, \qquad \mathbf{R}_k \in \mathbb{R}^{(k-d_k)\times k}.$$

Then

$$\|\mathbf{A}_k - \mathbf{Q}_k\mathbf{R}_k\|_F \;\le\; \varepsilon\, d_k\, \|\mathbf{R}_k\|_F.$$

Note that one can monitor this error bound as the method progresses.

Incremental One-Pass QR Factorization: Analysis

How does badly does this simple truncation strategy compromise the accuracy of the factorization?

Let $\mathbf{A}_k = \mathbf{A}(:, 1:k)$ denote the first k columns of $\mathbf{A}$.

Theorem. Perform k steps of the incremental QR algorithm to get $\mathbf{A}_k \approx \mathbf{Q}_k\mathbf{R}_k$ using d_k deletions governed by the tolerance ε:

$$\mathbf{A}_k \in \mathbb{R}^{n\times k}, \qquad \mathbf{Q}_k \in \mathbb{R}^{n\times(k-d_k)}, \qquad \mathbf{R}_k \in \mathbb{R}^{(k-d_k)\times k}.$$

Then

$$\|\mathbf{A}_k - \mathbf{Q}_k\mathbf{R}_k\|_F \;\le\; \varepsilon\, d_k\, \|\mathbf{R}_k\|_F.$$

Note that one can monitor this error bound as the method progresses.

Corollary. Suppose $\mathbf{A} \approx \widehat{\mathbf{Q}}\widehat{\mathbf{R}}$ has been computed via the incremental QR algorithm with d deletions and tolerance ε. Let $\widehat{\mathbf{R}} = \widehat{\mathbf{V}}\mathbf{\Sigma}\mathbf{W}^*$ be an SVD of $\widehat{\mathbf{R}}$. Then $\big(\widehat{\mathbf{Q}}\widehat{\mathbf{V}}\big)\mathbf{\Sigma}\mathbf{W}^*$ is an approximate SVD of $\mathbf{A}$ with

$$\|\mathbf{A} - \big(\widehat{\mathbf{Q}}\widehat{\mathbf{V}}\big)\mathbf{\Sigma}\mathbf{W}^*\|_F \le \varepsilon\, d\, \|\widehat{\mathbf{R}}\|_F.$$

Thus we have an approximate SVD of $\mathbf{A}$ with controllable accuracy in one pass through the data.

Analysis of the CUR Approximations

How close can a rank-k CUR factorization come to the optimal approximation?

$$\|\mathbf{A} - \mathbf{V}_k \mathbf{\Sigma}_k \mathbf{W}_k^{\mathsf{T}}\| = \sigma_{k+1}$$

Analysis of the CUR Approximations

How close can a rank-k CUR factorization come to the optimal approximation?

$$\|\mathbf{A} - \mathbf{V}_k \mathbf{\Sigma}_k \mathbf{W}_k^\top\| = \sigma_{k+1}$$

Any row/comlumn selection scheme gives $\mathbf{C} = \mathbf{AF}$ and $\mathbf{R} = \mathbf{E}^\top \mathbf{A}$, so the analysis that follows applies to any CUR factorization [Ipsen].

Analysis of CUR Factorizations: Step 1

Step 1: Triangle inequality splits the error into row and column projections.

To analyze the accuracy of a CUR factorization with $\mathbf{U} = \mathbf{C}^{+}\mathbf{A}\mathbf{R}^{+}$, begin by splitting the problem into estimates for two orthogonal projections [Mahoney & Drineas 2009].

Here $\|\cdot\|$ represents the matrix 2-norm.

$$\begin{aligned}
\|\mathbf{A} - \mathbf{CUR}\| &= \|\mathbf{A} - \mathbf{CC}^{+}\mathbf{AR}^{+}\mathbf{R}\| \\
&= \|\mathbf{A} - \mathbf{CC}^{+}\mathbf{A} + \mathbf{CC}^{+}\mathbf{A} - \mathbf{CC}^{+}\mathbf{AR}^{+}\mathbf{R}\| \\
&\le \|(\mathbf{I} - \mathbf{CC}^{+})\mathbf{A}\| + \|\mathbf{CC}^{+}\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\| \\
&\le \|(\mathbf{I} - \mathbf{CC}^{+})\mathbf{A}\| + \|\mathbf{CC}^{+}\|\,\|\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\| \\
&= \|(\mathbf{I} - \mathbf{CC}^{+})\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\|,
\end{aligned}$$

since $\mathbf{CC}^{+}$ is an orthogonal projector.

Analysis of CUR Factorizations: Step 2

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: Introduce a superfluous projector to set up a later inequality.

We shall focus on the $\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$; the other term is similar.

- Since $\mathbf{R} \in \mathbb{R}^{k\times n}$ with $k \le n$, its pseudoinverse is $\mathbf{R}^+ = \mathbf{R}^\mathsf{T}(\mathbf{R}\mathbf{R}^\mathsf{T})^{-1}$.
- Recall the interpolatory projector $\mathcal{P} = \mathbf{V}(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\mathbf{E}^\mathsf{T}$.
- In this setting, one can show: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$.

Analysis of CUR Factorizations: Step 2

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: Introduce a superfluous projector to set up a later inequality.

We shall focus on the $\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$; the other term is similar.

- Since $\mathbf{R} \in \mathbb{R}^{k\times n}$ with $k \le n$, its pseudoinverse is $\mathbf{R}^+ = \mathbf{R}^T(\mathbf{RR}^T)^{-1}$.
- Recall the interpolatory projector $\mathcal{P} = \mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\mathbf{E}^T$.
- In this setting, one can show: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$.

Proof: Write

$$\begin{aligned}
\mathcal{P}\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) &= \mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\mathbf{E}^T\mathbf{A}(\mathbf{I} - \mathbf{R}^T(\mathbf{RR}^T)^{-1}\mathbf{R}) \\
&= \mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\,\mathbf{R}\,(\mathbf{I} - \mathbf{R}^T(\mathbf{RR}^T)^{-1}\mathbf{R}) \\
&= \mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}(\mathbf{R} - \mathbf{RR}^T(\mathbf{RR}^T)^{-1}\mathbf{R}) \\
&= \mathbf{0},
\end{aligned}$$

and hence

$$\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}).$$

Analysis of CUR Factorizations: Step 3

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$

Step 3: Bound $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

$$\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \;=\; \|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$$

Analysis of CUR Factorizations: Step 3

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$

Step 3: Bound $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

$$\begin{aligned}\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| &= \|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \\ &\le \|(\mathbf{I} - \mathcal{P})\mathbf{A}\|\,\|\mathbf{I} - \mathbf{R}^+\mathbf{R}\|\end{aligned}$$

Analysis of CUR Factorizations: Step 3

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$

Step 3: Bound $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

$$\begin{aligned}
\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| &= \|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \\
&\le \|(\mathbf{I} - \mathcal{P})\mathbf{A}\| \, \|\mathbf{I} - \mathbf{R}^+\mathbf{R}\| \\
&= \|(\mathbf{I} - \mathcal{P})\mathbf{A}\|
\end{aligned}$$

Analysis of CUR Factorizations: Step 3

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^{+})\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})$

Step 3: Bound $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\|$

$$\begin{aligned}
\|\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\| &= \|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^{+}\mathbf{R})\| \\
&\le \|(\mathbf{I} - \mathcal{P})\mathbf{A}\|\,\|\mathbf{I} - \mathbf{R}^{+}\mathbf{R}\| \\
&= \|(\mathbf{I} - \mathcal{P})\mathbf{A}\|
\end{aligned}$$

Now recall that $\mathbf{\Pi} = \mathbf{V}\mathbf{V}^{T}$ is the orthogonal projector onto $\mathrm{Ran}(\mathbf{V})$.
Since $\mathcal{P}$ is the interpolatory projector onto $\mathrm{Ran}(\mathbf{V})$, $\mathcal{P}\mathbf{\Pi} = \mathbf{\Pi}$, and so

$$(\mathbf{I} - \mathcal{P})(\mathbf{I} - \mathbf{\Pi}) = \mathbf{I} - \mathcal{P}.$$

Analysis of CUR Factorizations: Step 3

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$

Step 3: Bound $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

$$\begin{aligned} \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| &= \|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \\ &\le \|(\mathbf{I} - \mathcal{P})\mathbf{A}\|\,\|\mathbf{I} - \mathbf{R}^+\mathbf{R}\| \\ &= \|(\mathbf{I} - \mathcal{P})\mathbf{A}\| \end{aligned}$$

Now recall that $\mathbf{\Pi} = \mathbf{V}\mathbf{V}^T$ is the orthogonal projector onto $\mathrm{Ran}(\mathbf{V})$.
Since $\mathcal{P}$ is the interpolatory projector onto $\mathrm{Ran}(\mathbf{V})$, $\mathcal{P}\mathbf{\Pi} = \mathbf{\Pi}$, and so

$$(\mathbf{I} - \mathcal{P})(\mathbf{I} - \mathbf{\Pi}) = \mathbf{I} - \mathcal{P}.$$

$$\begin{aligned} \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \le \|(\mathbf{I} - \mathcal{P})\mathbf{A}\| &= \|(\mathbf{I} - \mathcal{P})(\mathbf{I} - \mathbf{\Pi})\mathbf{A}\| \\ &\le \underbrace{\|\mathbf{I} - \mathcal{P}\|}_{\substack{\text{obliquity of the} \\ \text{interpolatory projector}}} \; \underbrace{\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A}\|}_{\substack{\text{accuracy of the} \\ \text{singular vectors } \mathbf{V}}} \end{aligned}$$

Analysis of CUR Factorizations: Step 4

Step 1: $\|\mathbf{A}-\mathbf{CUR}\| \le \|(\mathbf{I}-\mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I}-\mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I}-\mathbf{R}^+\mathbf{R}) = (\mathbf{I}-\mathcal{P})\mathbf{A}(\mathbf{I}-\mathbf{R}^+\mathbf{R})$

Step 3: $\|(\mathbf{I}-\mathcal{P})\mathbf{A}(\mathbf{I}-\mathbf{R}^+\mathbf{R})\| \le \|\mathbf{I}-\mathcal{P}\|\,\|(\mathbf{I}-\mathbf{\Pi})\mathbf{A})\|$

Step 4: Bound $\|\mathbf{I}-\mathcal{P}\|$ and $\|(\mathbf{I}-\mathbf{\Pi})\mathbf{A}\|$

Since $\mathcal{P}$ is a projector (assuming $\mathcal{P} \neq \mathbf{0}, \mathbf{I}$), we have $\|\mathbf{I}-\mathcal{P}\| = \|\mathcal{P}\|$, so

$$\begin{aligned}\|\mathbf{I}-\mathcal{P}\| = \|\mathcal{P}\| &= \|\mathbf{V}(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\mathbf{E}^\mathsf{T}\| \\ &\le \|\mathbf{V}\|\,\|(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\|\,\|\mathbf{E}^\mathsf{T}\| \\ &= \|(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\|.\end{aligned}$$

This value $\|\mathbf{I}-\mathcal{P}\|$ is the Lebesgue constant for the discrete interpolation.

Analysis of CUR Factorizations: Step 4

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$

Step 3: $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \le \|\mathbf{I} - \mathcal{P}\|\,\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A})\|$

Step 4: Bound $\|\mathbf{I} - \mathcal{P}\|$ and $\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A}\|$

Since $\mathcal{P}$ is a projector (assuming $\mathcal{P} \neq \mathbf{0}, \mathbf{I}$), we have $\|\mathbf{I} - \mathcal{P}\| = \|\mathcal{P}\|$, so

$$\begin{aligned} \|\mathbf{I} - \mathcal{P}\| = \|\mathcal{P}\| &= \|\mathbf{V}(\mathbf{E}^T\mathbf{V})^{-1}\mathbf{E}^T\| \\ &\le \|\mathbf{V}\|\,\|(\mathbf{E}^T\mathbf{V})^{-1}\|\,\|\mathbf{E}^T\| \\ &= \|(\mathbf{E}^T\mathbf{V})^{-1}\|. \end{aligned}$$

This value $\|\mathbf{I} - \mathcal{P}\|$ is the Lebesgue constant for the discrete interpolation.

When $\mathbf{V}$ contains the exact leading k singular vectors,

$$\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A}\| = \sigma_{k+1},$$

thus giving

$$\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \le \|(\mathbf{E}^T\mathbf{V})^{-1}\|\,\sigma_{k+1}.$$

cf. [Halko, Martinsson, Tropp, 2011; Ipsen]

Analysis of CUR Factorizations: Summary

Step 1: $\|\mathbf{A} - \mathbf{CUR}\| \le \|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| + \|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\|$.

Step 2: $\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R}) = (\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})$.

Step 3: $\|(\mathbf{I} - \mathcal{P})\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \le \|\mathbf{I} - \mathcal{P}\|\,\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A})\|$.

Step 4: Bound $\|\mathbf{I} - \mathcal{P}\|\,\|(\mathbf{I} - \mathbf{\Pi})\mathbf{A}\| \le \|(\mathbf{E}^T\mathbf{V})^{-1}\|\,\sigma_{k+1}$.

In summary,

$$\|\mathbf{A}(\mathbf{I} - \mathbf{R}^+\mathbf{R})\| \le \|(\mathbf{E}^T\mathbf{V})^{-1}\|\,\sigma_{k+1}.$$

Similarly, for the column projection,

$$\|(\mathbf{I} - \mathbf{CC}^+)\mathbf{A}\| \le \|(\mathbf{W}^T\mathbf{F})^{-1}\|\,\sigma_{k+1}.$$

Putting these pieces together,

$$\|\mathbf{A} - \mathbf{CUR}\| \le \Big(\|(\mathbf{E}^T\mathbf{V})^{-1}\| + \|(\mathbf{W}^T\mathbf{F})^{-1}\|\Big)\,\sigma_{k+1}.$$

Analysis of DEIM-CUR Factorization

Theorem. Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ and $\mathbf{W} \in \mathbb{R}^{n\times k}$ contain the k leading left and right singular vectors of $\mathbf{A}$, and let $\mathbf{E} = \mathbf{I}(:,\mathbf{p})$ and $\mathbf{F} = \mathbf{I}(:,\mathbf{q})$ for $\mathbf{p} = \mathrm{DEIM}(\mathbf{V})$ and $\mathbf{q} = \mathrm{DEIM}(\mathbf{W})$. Then for $\mathbf{C} = \mathbf{AF}$, $\mathbf{R} = \mathbf{E}^{\mathsf{T}}\mathbf{A}$, and $\mathbf{U} = \mathbf{C}^{+}\mathbf{A}\mathbf{R}^{+}$,

$$\|\mathbf{A} - \mathbf{CUR}\| \leq \Big(\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\| + \|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\|\Big)\sigma_{k+1}.$$

Analysis of DEIM-CUR Factorization

Theorem. Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ and $\mathbf{W} \in \mathbb{R}^{n\times k}$ contain the k leading left and right singular vectors of $\mathbf{A}$, and let $\mathbf{E} = \mathbf{I}(:, \mathbf{p})$ and $\mathbf{F} = \mathbf{I}(:, \mathbf{q})$ for $\mathbf{p} = \mathrm{DEIM}(\mathbf{V})$ and $\mathbf{q} = \mathrm{DEIM}(\mathbf{W})$. Then for $\mathbf{C} = \mathbf{AF}$, $\mathbf{R} = \mathbf{E}^{\mathsf{T}}\mathbf{A}$, and $\mathbf{U} = \mathbf{C}^{+}\mathbf{A}\mathbf{R}^{+}$,

$$\|\mathbf{A} - \mathbf{CUR}\| \le \Big(\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\| + \|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\|\Big)\sigma_{k+1}.$$

Lemma. [Chaturantabut, Sorensen 2010]

$$\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\| \le \frac{(1+\sqrt{2n})^{k-1}}{\|\mathbf{w}_1\|_\infty}, \qquad \|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\| \le \frac{(1+\sqrt{2m})^{k-1}}{\|\mathbf{v}_1\|_\infty}.$$

Analysis of DEIM-CUR Factorization

Theorem. Let $\mathbf{V} \in \mathbb{R}^{m\times k}$ and $\mathbf{W} \in \mathbb{R}^{n\times k}$ contain the k leading left and right singular vectors of $\mathbf{A}$, and let $\mathbf{E} = \mathbf{I}(:,\mathbf{p})$ and $\mathbf{F} = \mathbf{I}(:,\mathbf{q})$ for $\mathbf{p} = \mathrm{DEIM}(\mathbf{V})$ and $\mathbf{q} = \mathrm{DEIM}(\mathbf{W})$. Then for $\mathbf{C} = \mathbf{AF}$, $\mathbf{R} = \mathbf{E}^{\mathsf{T}}\mathbf{A}$, and $\mathbf{U} = \mathbf{C}^{+}\mathbf{A}\mathbf{R}^{+}$,

$$\|\mathbf{A} - \mathbf{CUR}\| \le \Big(\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\| + \|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\|\Big)\sigma_{k+1}.$$

Lemma. Improved DEIM error bound:

$$\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\| < \sqrt{\frac{nk}{3}}\,2^k, \qquad \|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\| < \sqrt{\frac{mk}{3}}\,2^k.$$

- Compare to analogous bound by Drmač and Gugercin for Q-DEIM.
- One can construct an example with $O(2^k)$ growth.
- Like Gaussian Elimination with partial pivoting, the worst-case growth factor is exponential in k, but performance is much better in practice.
- To analyze other row/column selection schemes, one only needs to bound $\|(\mathbf{W}^{\mathsf{T}}\mathbf{F})^{-1}\|$ and $\|(\mathbf{E}^{\mathsf{T}}\mathbf{V})^{-1}\|$ for the given method.

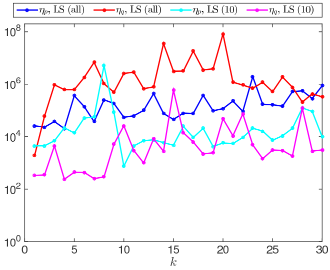
Some Examples of the DEIM-CUR Approximation

Example: Sparse + Steady Singular Value Decay

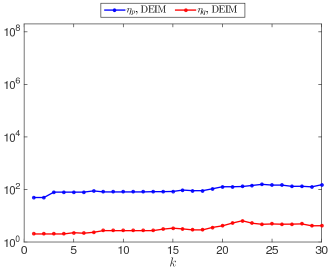
Consider a sparse matrix constructed to have steady singular value decay, with a gap: $\mathbf{A} \in \mathbb{R}^{m \times n}$ for $m = 300{,}000$ and $n = 300$:

$$\mathbf{A} = \sum_{j=1}^{10} \frac{2}{j} \mathbf{x}_j \mathbf{y}_j^T + \sum_{j=11}^{300} \frac{1}{j} \mathbf{x}_j \mathbf{y}_j^T .$$

$$\boxed{\|\mathbf{A} - \mathbf{CUR}\| \le \left(\|(\mathbf{W}^\mathsf{T}\mathbf{F})^{-1}\| + \|(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\| \right) \sigma_{k+1}}$$



Leverage Scores (all and 10 sv's)



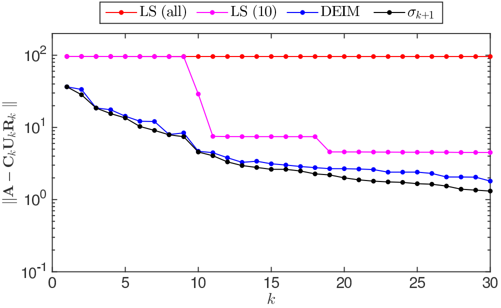
DEIM

Example: Sparse + Steady Singular Value Decay

Consider a sparse matrix constructed to have steady singular value decay, with a gap: $\mathbf{A} \in \mathbb{R}^{m \times n}$ for $m = 300{,}000$ and $n = 300$:

$$\mathbf{A} = \sum_{j=1}^{10} \frac{2}{j}\, \mathbf{x}_j \mathbf{y}_j^T + \sum_{j=11}^{300} \frac{1}{j}\, \mathbf{x}_j \mathbf{y}_j^T .$$

$$\boxed{\|\mathbf{A} - \mathbf{CUR}\| \le \left(\|(\mathbf{W}^T\mathbf{F})^{-1}\| + \|(\mathbf{E}^T\mathbf{V})^{-1}\| \right) \sigma_{k+1}}$$



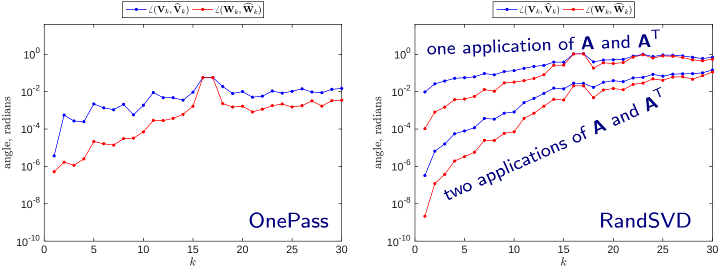
Example: Sparse + Steady Singular Value Decay

How do inaccurate singular vectors have on the DEIM-CUR factorization?

Approximate the SVD via OnePass QR method (tolerance 10^{-4}) and RandSVD [Halko, Matrinsson, Tropp, 2011] (cf. subspace iteration on a random block of vectors) *with only one or two applications of* $\mathbf{A}$ *and* $\mathbf{A}^T$.

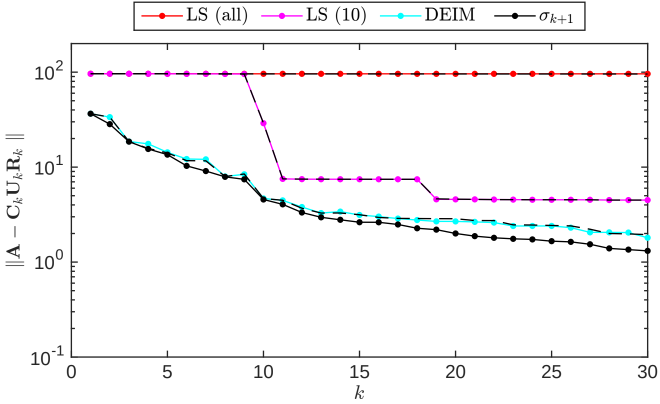
$\mathbf{V}_k =$ "exact" leading k singular vectors
$\widehat{\mathbf{V}}_k =$ leading k singular vectors from randSVD

Largest canonical angle between the subspaces:



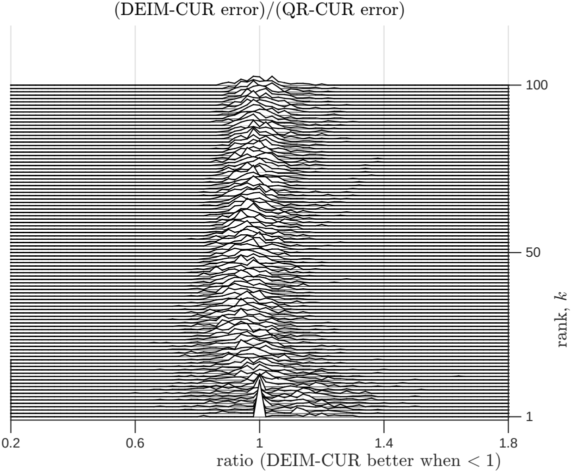
Example: Sparse + Steady Singular Value Decay

The "dirty" singular vectors have very little effect on the accuracy of the DEIM approximation. In the plot below, the inexact singular vectors from RandSVD (*one application of* $\mathbf{A}$ *and* $\mathbf{A}^T$) are shown as the dashed black line.



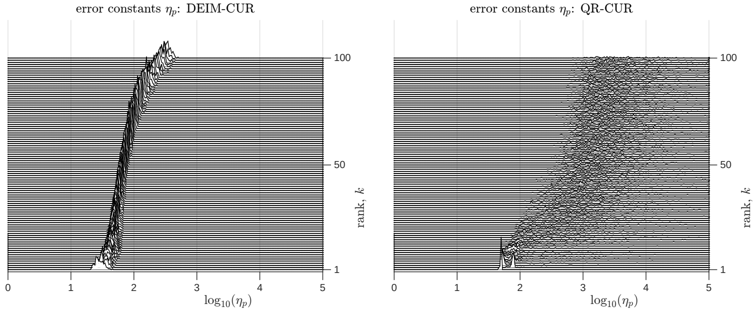
Example: Sparse + Steady Singular Value Decay

DEIM-CUR accuracy is typically similar to CUR derived from column-pivoted QR, but gives smaller error constants. A comparison of 100 random trials:



Example: Sparse + Steady Singular Value Decay

DEIM-CUR accuracy is typically similar to CUR derived from column-pivoted QR, but gives smaller error constants. A comparison of 100 random trials:



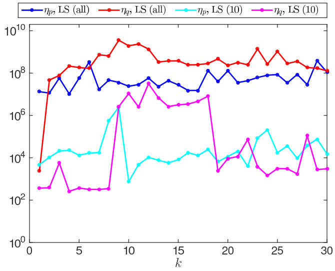
Example: Sparse + Steady Singular Value Decay

Consider a sparse matrix constructed to have steady singular value decay, with a big gap: $\mathbf{A} \in \mathbb{R}^{m\times n}$ for $m = 300{,}000$ and $n = 300$:

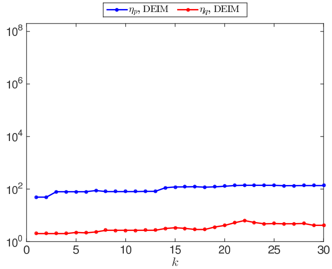
$$\mathbf{A} = \sum_{j=1}^{10} \frac{1000}{j}\,\mathbf{x}_j\mathbf{y}_j^T + \sum_{j=11}^{300} \frac{1}{j}\,\mathbf{x}_j\mathbf{y}_j^T.$$

Error bound for CUR factorizations:

$$\boxed{\|\mathbf{A} - \mathbf{CUR}\| \le \left(\|(\mathbf{W}^\mathsf{T}\mathbf{F})^{-1}\| + \|(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\|\right)\sigma_{k+1}}$$



Leverage Scores (all and 10 sv's)



DEIM

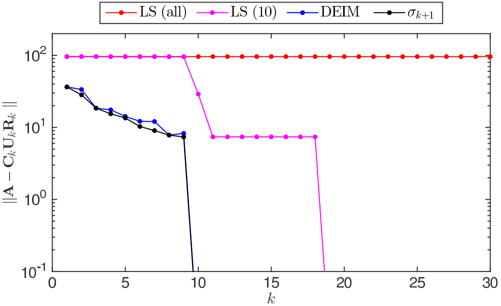
Example: Sparse + Steady Singular Value Decay

Consider a sparse matrix constructed to have steady singular value decay, with a big gap: $\mathbf{A} \in \mathbb{R}^{m \times n}$ for $m = 300{,}000$ and $n = 300$:

$$\mathbf{A} = \sum_{j=1}^{10} \frac{1000}{j}\, \mathbf{x}_j \mathbf{y}_j^T + \sum_{j=11}^{300} \frac{1}{j}\, \mathbf{x}_j \mathbf{y}_j^T .$$

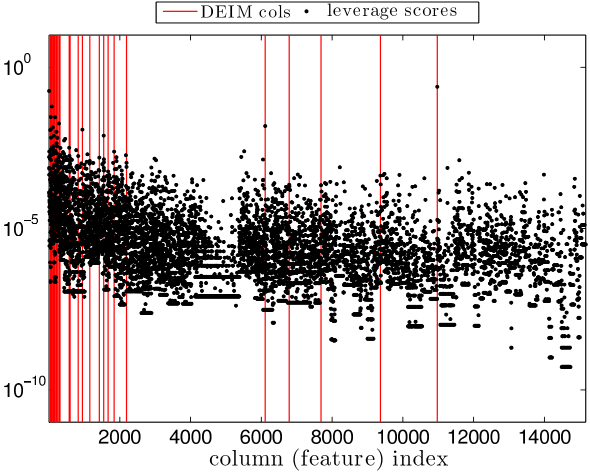
Error bound for CUR factorizations:

$$\boxed{\|\mathbf{A} - \mathbf{CUR}\| \le \left(\|(\mathbf{W}^\mathsf{T}\mathbf{F})^{-1}\| + \|(\mathbf{E}^\mathsf{T}\mathbf{V})^{-1}\| \right) \sigma_{k+1}}$$



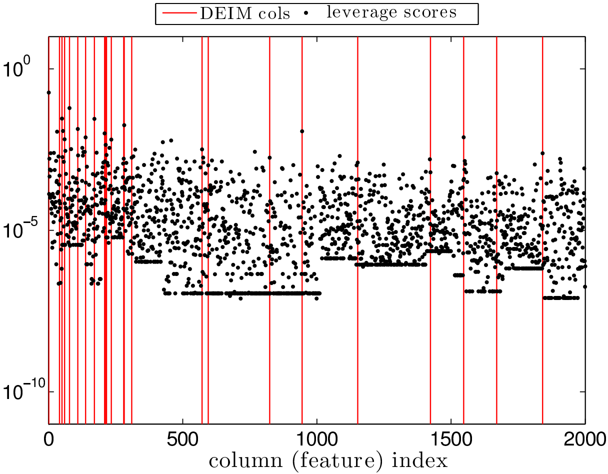
Term document example

A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139 \times 15170}$, $k = 30$



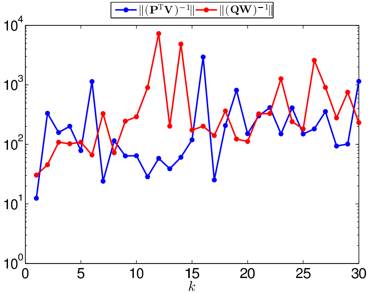
Term document example

A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139\times 15170}$, $k = 30$

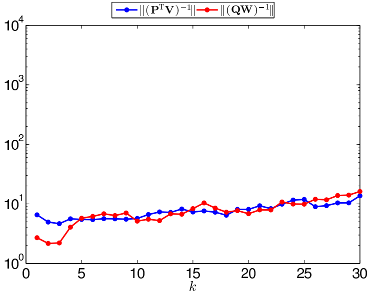


Term document example

A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139 \times 15170}$, $k = 30$



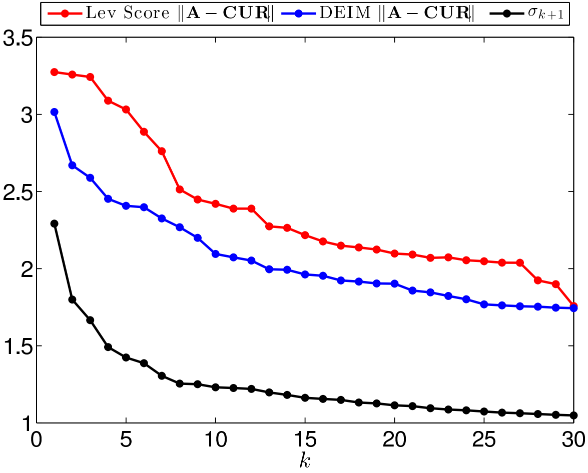
Leverage Scores



DEIM

Term document example

A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139\times 15170}$, $k = 30$



Term document example

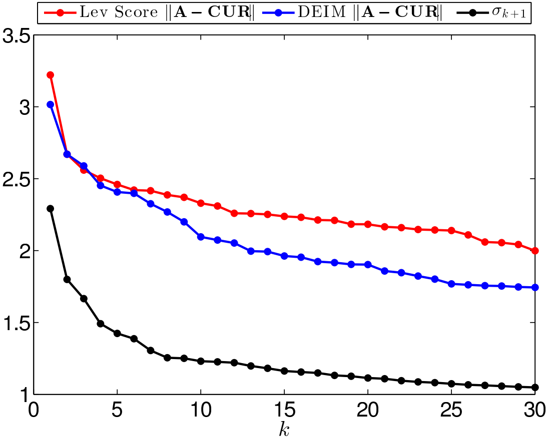
A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139\times 15170}$, $k = 30$

Comparison of DEIM columns with those of leverage scores (LS) using *all* singular vectors versus only *two* leading singular vectors.
(The leverage scores are normalized.)

DEIM rank, j	index, q_j	LS (all)	LS (2)	term
1	10973	0.875	1.000	evansville
2	1	0.726	0.741	florida
3	1547	0.948	0.031	spacer
4	109	0.347	0.055	contact
5	209	0.458	0.040	service
6	50	0.739	0.116	miami
7	824	0.809	0.007	chapter
8	1841	0.537	0.010	health
9	171	0.617	0.113	information
10	234	0.436	0.026	events

Term document example

A data set from the TechTC collection, used by [Mahoney, Drineas 2009].
Concatenation of web pages about Evansville, Indiana and Miami, Florida.
$\mathbf{A} \in \mathbb{R}^{139 \times 15170}$, $k = 30$



CUR error for leverage scores based only on the *two leading singular vectors*.

Gait Analysis from Building Vibrations

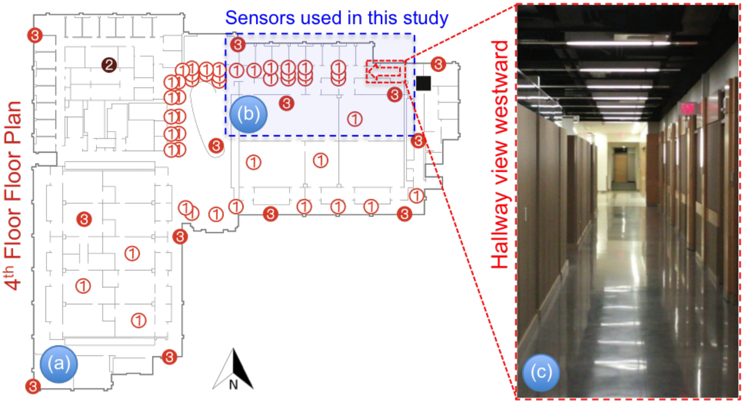
w/Dustin Bales, Serkan Gugercin, Rodrigo Sarlo, Pablo Tarazaga, and Mico Woolard

The Virginia Tech Smart Infrastructure Laboratory (VTSIL), founded by Dr. Pablo Tarazaga, has instrumented the campus's new Goodwin Hall with 212 accelerometers welded to the frame of the building to measure high-fidelity building vibrations. Applications include structural health monitoring, energy efficiency (e.g. HVAC), threat identification, and building evacuation assistance.

Gait Analysis from Building Vibrations

w/Dustin Bales, Serkan Gugercin, Rodrigo Sarlo, Pablo Tarazaga, and Mico Woolard

Proof of concept project: Send a walker down a heavily instrumented hallway. Can one classify the gender or weight of the walker based on vibrations?



Gait Analysis from Building Vibrations

w/Dustin Bales, Serkan Gugercin, Rodrigo Sarlo, Pablo Tarazaga, and Mico Woolard

Proof of concept project: Send a walker down a heavily instrumented hallway. Can one classify the gender and weight of the walker based on vibrations?

- Each footstep initiates vibrations in all the sensors.
- Vibrations detected by the various sensors show significant redundancy.
- Can we identify a minimal set of independent sensors? (Cf. sensor placement; deploying a smaller array of sensors in other buildings, etc.)

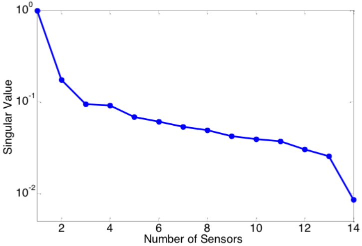
Gait Analysis from Building Vibrations

w/Dustin Bales, Serkan Gugercin, Rodrigo Sarlo, Pablo Tarazaga, and Mico Woolard

Proof of concept project: Send a walker down a heavily instrumented hallway. Can one classify the gender and weight of the walker based on vibrations?

- Each footstep initiates vibrations in all the sensors.
- Vibrations detected by the various sensors show significant redundancy.
- Can we identify a minimal set of independent sensors? (Cf. sensor placement; deploying a smaller array of sensors in other buildings, etc.)

- Singular values of representative data matrix: (number of measurements) $\times$ (number of sensors)



Gait Analysis from Building Vibrations

w/Dustin Bales, Serkan Gugercin, Rodrigo Sarlo, Pablo Tarazaga, and Mico Woolard

Proof of concept project: Send a walker down a heavily instrumented hallway. Can one classify the gender and weight of the walker based on vibrations?

- Each footstep initiates vibrations in all the sensors.
- Vibrations detected by the various sensors show significant redundancy.
- Can we identify a minimal set of independent sensors? (Cf. sensor placement; deploying a smaller array of sensors in other buildings, etc.)

- CUR factorizations are used to find an independent set of sensors.
- $\|(\mathbf{P}^T\mathbf{V})^{-1}\|$ bound on Lebesgue constant informs sensor selection.
- The rankings of many trials are aggregated.
- The top sensors are then used for the data classification task.

Summary

- Low-rank CUR approximations capture properties of the data set.
- DEIM selection strategy gives column/row selection for CUR
- The SVD can be approximated using an incremental one-pass QR factorization or RandSVD.
- Error bound for general case ($\mathbf{U} = \mathbf{C}^{+}\mathbf{A}\mathbf{R}^{+}$)
  It would be nice to better characterize $\|(\mathbf{E}^{T}\mathbf{V})^{-1}\|$ for DEIM, e.g. average case analysis.
- In examples, DEIM-CUR is effective at reducing $\|\mathbf{A} - \mathbf{CUR}\|$.