Approximating Matrices: One Picture and One Thousand Words

Gene Golub Memorial Day, February 2017

Dianne P. O'Leary

In Honor of Gene H. Golub
Approximating Matrices:
One Picture and One Thousand Words

Dianne P. O'Leary

Computer Science Dept. and
Institute for Advanced Computer Studies
University of Maryland

Joint work with
Julianne Chung,
Matthias Chung,
John M. Conroy,
Yi-Kai Liu

Support from NSF.

Introduction: The Problem

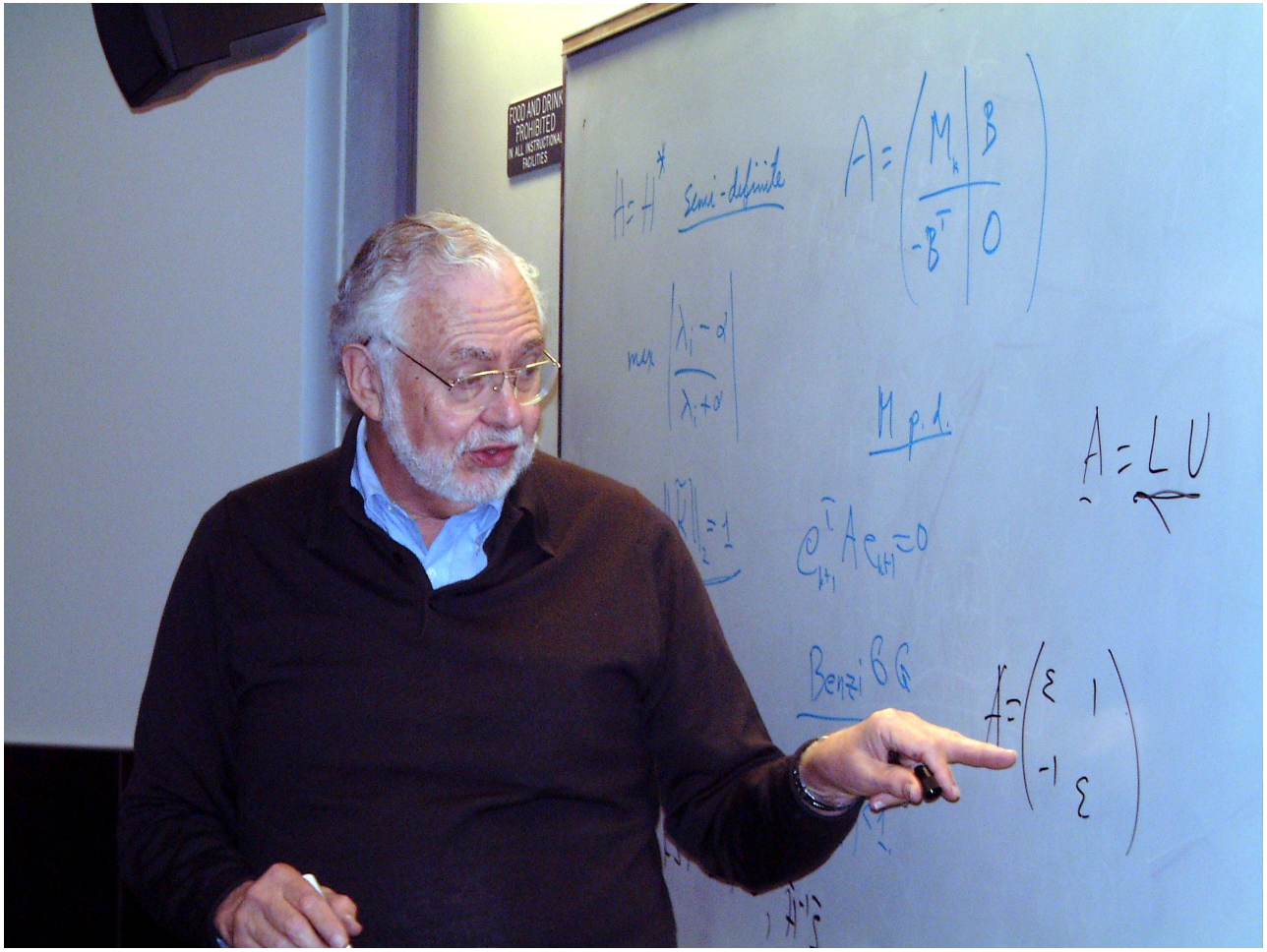
Given: a rank r matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, with $m \geq n$,

Determine: an approximation $\boldsymbol{Z}$ of rank $k \leq r$.

Low rank approximations of matrices and their inverses play an important role in many applications, such as image reconstruction, machine learning, text processing, matrix completion problems, signal processing, optimal control problems, statistics, and mathematical biology.

This work was inspired by Gene Golub

An inspired and inspiring teacher,



A dedicated mentor,

A generous friend to many of us here,

An outstanding researcher and catalyst.

Major contributions to

- numerical solution of eigenvalue problems,
- solution of least squares and total least squares problems,
- iterative solution of linear systems,
- understanding of orthogonal polynomials and quadrature,
- computation and applications of matrix factorizations.

Low-rank approximation was a focus of Gene's work for over 40 years

From his post-doctoral work, partially motivated by the use of the SVD in low-rank matrix approximation...

Calculating the Singular Values and Pseudo-Inverse of a Matrix
Gene Golub and W Kahan, SIAM Journal 1965

... To one of his last publications:

Rank-one Approximation to High Order Tensors
T Zhang, GH Golub, SIMAX 2001

With many important contributions in between, such as:

- Tracking a Few Extreme Singular Values and Vectors in Signal Processing, P Comon, GH Golub - Proceedings of the IEEE, 1990
  Maintaining a low-rank approximation to a covariance matrix slowly changing over time, based on Gene's updating methods for matrix factorizations.
- Fast Algorithms for Updating Signal Subspaces, G Xu, H Zha, G Golub, T Kailath - IEEE Transactions on Circuits, 1994
  Tracking a low-dimensional subspace slowly changing over time using the Lanczos algorithm.
- The Restricted Singular Value Decomposition: Properties and Applications, Bart LR De Moor and Gene H Golub, SIMAX 2006
  Low-rank approximations to partitioned matrices.

Gene emphasized two methods for computing low-rank approximations

- The Singular Value Decomposition (SVD)
- The Conjugate Gradient / Lanczos algorithm

Low-rank approximations from the SVD: best in the Frobenius norm

$$\min_{\text{rank}(\boldsymbol{Z})\leq k} f(\boldsymbol{A},\boldsymbol{Z}), \qquad f(\boldsymbol{A},\boldsymbol{Z}) = \|\boldsymbol{A}-\boldsymbol{Z}\|_F^2$$

Solution: best rank k approximation of an $m \times n$ matrix $\boldsymbol{A}$ Let

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\,\boldsymbol{V}^\top$$

be the singular value decomposition of $\boldsymbol{A}$, with $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n]$ and $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$.

The Eckart-Young Theorem, Schmidt 1907, Eckart and Young 1936, Mirsky 1960 (unitarily invariant norms)

$$\widehat{\boldsymbol{Z}} = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\,\boldsymbol{V}_k^\top$$

with

$$\boldsymbol{U}_k = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k], \quad \boldsymbol{V}_k = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k],$$
$$\boldsymbol{\Sigma}_k = \text{diag}(\sigma_1, \ldots, \sigma_k)\,.$$

The solution is unique if and only if $\sigma_k > \sigma_{k+1}$.

Low-rank approximations from conjugate gradients

In the 1975 Concus-Golub-O'Leary paper on the conjugate gradient algorithm, Gene made sure that the matrix approximation viewpoint was included: If an $n \times n$ symmetric positive definite matrix $\boldsymbol{A}$ could be expressed as

$$\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N},$$

where $\boldsymbol{M}$ is a preconditioner (close to $\boldsymbol{A}$ in some sense and easily inverted), then the matrix

$$\boldsymbol{K} = \boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{N} = \bar{\boldsymbol{Z}}_n \boldsymbol{J}_n \bar{\boldsymbol{Z}}_n^{-1},$$

where

- the columns of $\bar{\boldsymbol{Z}}_n$ are the preconditioned residuals and are $\boldsymbol{M}$-orthogonal,
- $\boldsymbol{J}_n$ is a tridiagonal matrix containing the cg parameters.

Therefore, after $k < n$ steps, we obtain a matrix approximation

$$\boldsymbol{K} \approx \bar{\boldsymbol{Z}}_k \boldsymbol{J}_k \bar{\boldsymbol{Z}}_k^{\dagger}.$$

Is that the end of the story, or just the beginning?

In approximating $\boldsymbol{A}$ by $\boldsymbol{Z}$, how we

- choose $f(\boldsymbol{A}, \boldsymbol{Z})$ to measure the goodness of the approximation, and
- what additional constraints we put on $\boldsymbol{Z}$

determine how hard our problem is.

How hard can it be?

How hard can it be?

Today we'll consider two other ways to form low-rank approximations:

- Inverse approximation.
  Joint work with Julianne Chung and Matthias Chung.
- Interval approximation.
  Joint work with John M. Conroy and Yi-Kai Liu.

And we'll discuss why these viewpoints are necessary and useful.

Just as in the Eckart-Young theorem, our approximations take the form

$$\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{Y}, \qquad \boldsymbol{X} : m \times k,\ \boldsymbol{Y} : k \times n.$$

First we'll give two examples of why we might want constraints on $\boldsymbol{X}$ and $\boldsymbol{Y}$, not just on $\boldsymbol{Z}$.

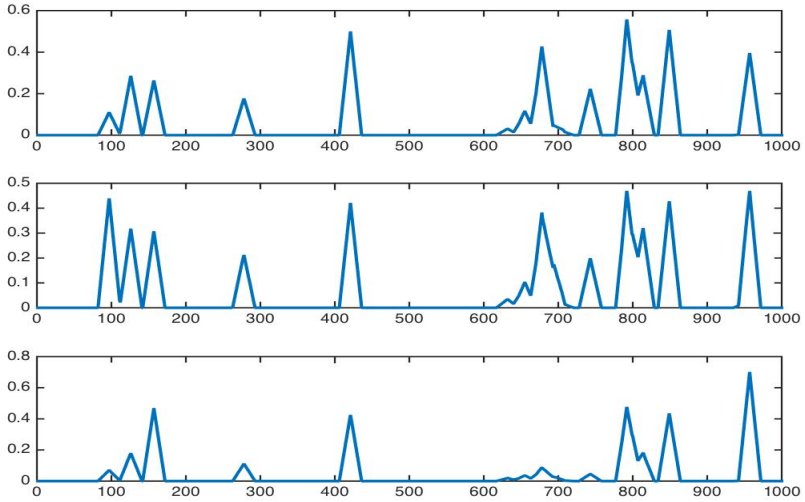
This seems odd.

Why put constraints on $\boldsymbol{X}$ and $\boldsymbol{Y}$ when it is $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{Y}$ that is supposed to approximate $\boldsymbol{A}$?

An application: spectroscopy

Suppose we are given 50 samples, each made up of various proportions of 5 unknown substances that don't react with each other.

And suppose we use spectroscopy (IR, for example) to measure a (noisy) spectrum ($m = 1000$ numbers) for each sample. Here are 3 of the 50 (simulated data):



These are 3 columns of an $m \times 50$ data matrix $\boldsymbol{A}$.

Each spectrum measures a response at 1000 frequencies, and they are nonnegative.

We've been told that each spectrum $\boldsymbol{a}_i$ is of the form

$$\boldsymbol{a}_i = y_{1i}\boldsymbol{x}_1 + y_{2i}\boldsymbol{x}_2 + y_{3i}\boldsymbol{x}_3 + y_{4i}\boldsymbol{x}_4 + y_{5i}\boldsymbol{x}_5, \qquad i = 1, \ldots, n$$

$\boldsymbol{x}$ vectors: the spectra for the five unknown substances
y coefficients: how much of each substance is in the ith sample.

So we want a good approximation $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{Y}$ to the data matrix $\boldsymbol{A}$, but to make it physically meaningful, we need both $\boldsymbol{X}$ and $\boldsymbol{Y}$ to be nonnegative.

Also, the vectors $\boldsymbol{X}$ should be sparse, because the spectrum for each substance has a small number of peaks (nonzero responses).

Another application: document classification

Suppose we are given 500 documents (e.g., articles from the math literature) and we want to classify them into k subjects.

Let's gather the distinct terms in the documents ('decomposition', 'topology', 'category theory', 'derivative', etc.) and create a matrix $\boldsymbol{A}$ that has one row for each term.

Set a_{ij} to be a measure of the importance of term i in document j, for $i = 1, \ldots, m =$ the number of terms and $j = 1, \ldots, 500$.

If we can determine a nonnegative $m \times k$ matrix $\boldsymbol{X}$ and a nonnegative $k \times 500$ matrix $\boldsymbol{Y}$ with $\boldsymbol{A} \approx \boldsymbol{X}\boldsymbol{Y}$, then each column of $\boldsymbol{A}$ is approximately a combination of the columns of $\boldsymbol{X}$. The coefficients in the ith column of $\boldsymbol{Y}$ tell us how important each of these dictionary columns is to document i.

We would expect both $\boldsymbol{X}$ and $\boldsymbol{Y}$ to be nonnegative.

And we would also expect sparsity in each of these matrices.

Do we want to approximate a matrix, or its inverse?

A different measure of goodness: Inverse approximation

Instead of

$$\min_{\operatorname{rank}(\boldsymbol{Z})\le k} f(\boldsymbol{A},\boldsymbol{Z}), \qquad f(\boldsymbol{A},\boldsymbol{Z}) = \|\boldsymbol{A}-\boldsymbol{Z}\|_F^2$$

let's consider

$$\min_{\operatorname{rank}(\boldsymbol{Z})\le k} f(\boldsymbol{A},\boldsymbol{Z}) \qquad f(\boldsymbol{A},\boldsymbol{Z}) = \|\boldsymbol{Z}\boldsymbol{A}-\boldsymbol{I}_n\|_F^2$$

A solution:

$$\widehat{\boldsymbol{Z}} = \boldsymbol{V}_k \boldsymbol{\Sigma}_k^{-1} \boldsymbol{U}_k^\top.$$

But any choice of k nonzero singular values and the corresponding vectors gives a global minimizer, so the problem is not well-posed.

Adding constraints can make the problem easier.

For example, if we take our inverse approximation problem

$$\min_{\operatorname{rank}(\boldsymbol{Z})\leq k} f(\boldsymbol{A},\boldsymbol{Z}) \quad \text{where} \quad f(\boldsymbol{A},\boldsymbol{Z}) = \|\boldsymbol{Z}\boldsymbol{A}-\boldsymbol{I}_n\|_F^2,$$

and add the constraint

$$\|\boldsymbol{Z}\|_{\mathrm{F}} \leq c$$

for a small enough constant c, then we will see later that we have made the solution unique.

Adding constraints can make the problem harder.

Constraints on the factors $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{Y}$, where $\boldsymbol{X}$ is $m \times k$ and $\boldsymbol{Y}$ is $k \times n$ are useful but complicate the problem.

We could ask that

- $\boldsymbol{X}$ and $\boldsymbol{Y}$ be nonnegative, or
- $\boldsymbol{X}$ and $\boldsymbol{Y}$ be sparse.

Inverse Approximation: Finding a solution

$$\min_{\operatorname{rank}(\boldsymbol{Z})\leq k} \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_{\mathrm{F}}^2 + \alpha^2 \|\boldsymbol{Z}\|_{\mathrm{F}}^2 ,$$

- $\boldsymbol{A}$ has dimension $m \times n$, with $m \geq n$.
- $k \leq \operatorname{rank}(\boldsymbol{A})$ is a given positive integer.
- α is a given parameter, nonzero if $\operatorname{rank}(\boldsymbol{A}) < m$.
- $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$.

$$\min_{\text{rank}(\boldsymbol{Z})\leq k} \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_{\text{F}}^2 + \alpha^2 \|\boldsymbol{Z}\|_{\text{F}}^2 ,$$

Theorem: (Chung, Chung, O'Leary) A global minimizer $\widehat{\boldsymbol{Z}} \in \mathbb{R}^{n\times m}$ is

$$\widehat{\boldsymbol{Z}} = \boldsymbol{V}_k \boldsymbol{\Psi}_k \boldsymbol{U}_k^\top ,$$

where $\boldsymbol{V}_k$ contains the first k columns of $\boldsymbol{V}$, $\boldsymbol{U}_k$ contains the first k columns of $\boldsymbol{U}$, and

$$\boldsymbol{\Psi}_k = \text{diag}\left(\frac{\sigma_1}{\sigma_1^2 + \alpha^2}, \ldots, \frac{\sigma_k}{\sigma_k^2 + \alpha^2} \right) .$$

Moreover, if $\alpha \neq 0$, this $\widehat{\boldsymbol{Z}}$ is the *unique* global minimizer if and only if $\sigma_k > \sigma_{k+1}$

What if $\alpha = 0$?

$$\min_{\text{rank}(\boldsymbol{Z}) \leq k} \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_{\text{F}}^2 + \alpha^2 \|\boldsymbol{Z}\|_{\text{F}}^2 ,$$

A solution still exists, but, as noted by Friedland and Torokhti (2007), it is not unique if $k < \text{rank}(\boldsymbol{A})$. In fact,

$$\|\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n\|_{\text{F}}^2 \geq n - k.$$

We can achieve this lower bound by choosing $\boldsymbol{Z} = \boldsymbol{V}_k \boldsymbol{\Sigma}_k^{-1} \boldsymbol{U}_k^\top$, or a matrix of this form constructed using *any* choice of k singular values and corresponding singular vectors.

These are the only global minimizers.

What if we use a different norm?

Consider, for example,

$$\min_{\text{rank}(\boldsymbol{Z}) \leq k} \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_2^2 + \alpha^2 \|\boldsymbol{Z}\|_2^2 ,$$

where $\|\boldsymbol{F}\|_2^2$ is the largest eigenvalue of $\boldsymbol{F}^\top \boldsymbol{F}$.

- For any $\boldsymbol{Z}$ with rank $k < n$, $\|\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n\|_2^2 \geq 1$.
- Therefore, $\boldsymbol{Z} = \boldsymbol{0}$ is a global minimizer, unique if $\alpha \neq 0$, so the problem is not interesting.

How does $\boldsymbol{Z}$ relate to well-known approximate inverses $\boldsymbol{V\Phi\Sigma^{\dagger}U^{\top}}$?

$\boldsymbol{\Phi}$ is a diagonal matrix of filter factors ϕ_j.

- Tikhonov regularization:

$$\phi_j = \frac{\sigma_j^2}{\sigma_j^2 + \alpha^2}, \quad j = 1, \ldots, n.$$

- Truncated SVD (TSVD) regularization:

$$\phi_j = \begin{cases} 1, & \text{for } j \leq k, \\ 0, & \text{for } j > k. \end{cases}$$

How does $\boldsymbol{Z}$ relate to well-known approximate inverses $\boldsymbol{V}\boldsymbol{\Phi}\boldsymbol{\Sigma}^{\dagger}\boldsymbol{U}^{\top}$?

$\boldsymbol{\Phi}$ is a diagonal matrix of filter factors ϕ_j.

- Tikhonov regularization:

$$\phi_j = \frac{\sigma_j^2}{\sigma_j^2 + \alpha^2}, \quad j = 1, \ldots, n.$$

- Truncated SVD (TSVD) regularization:

$$\phi_j = \begin{cases} 1, & \text{for } j \leq k, \\ 0, & \text{for } j > k. \end{cases}$$

- Our approximate inverse: Truncated Tikhonov regularization

$$\phi_j = \begin{cases} \frac{\sigma_j^2}{\sigma_j^2+\alpha^2}, & \text{for } j \leq k, \\ 0, & \text{for } j > k. \end{cases}$$

How the problem arises in minimizing Bayes risk

Given

- an image $\boldsymbol{\xi}$, drawn according to a probability distribution of images,
- an operator $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ that distorts the observed image,
- noise $\boldsymbol{\delta}$, drawn according to a probability distribution of noise samples,
- an observation

$$\boldsymbol{b} = \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{\delta}.$$

It would be nice to have a matrix $\boldsymbol{Z}$ that minimizes error

$$\boldsymbol{Z}\boldsymbol{b} - \boldsymbol{\xi} = (\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\boldsymbol{\xi} + \boldsymbol{Z}\boldsymbol{\delta}$$

averaged over the sampling of images and noise.

This would allow fast reconstruction of the signals using the precomputed matrix $\boldsymbol{Z}$.

We choose to find $\boldsymbol{Z}$ by minimizing the *Bayes risk* $\tilde{f}(\boldsymbol{Z})$ defined, using our model

$$\boldsymbol{b} = \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{\delta}$$

and a quadratic loss function, to be

$$\tilde{f}(\boldsymbol{Z}) = \mathcal{E}\left(\|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\boldsymbol{\xi} + \boldsymbol{Z}\boldsymbol{\delta}\|_2^2\right),$$

where $\mathcal{E}$ denotes expected value.

Assumptions

Assume:

- The random variables $\boldsymbol{\xi}$ and $\boldsymbol{\delta}$ are statistically independent.
- The probability distribution for $\boldsymbol{\xi}$ has mean $\boldsymbol{\mu}_{\boldsymbol{\xi}}$ and variance $\eta^2 \boldsymbol{I}$.
- The probability distribution for $\boldsymbol{\delta}$ has mean $\boldsymbol{\mu}_{\boldsymbol{\delta}} = \mathbf{0}$ and covariance matrix $\beta^2 \boldsymbol{I}_m$.

(General covariance matrices can be handled.)

Minimizing the Bayes risk

Lemma: Under these assumptions, the Bayes risk is

$$\tilde{f}(\boldsymbol{Z}) = \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\boldsymbol{\mu}_{\boldsymbol{\xi}}\|_2^2 + \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_{\mathrm{F}}^2 + \alpha^2 \|\boldsymbol{Z}\|_{\mathrm{F}}^2 .$$

Theorem: Consider the problem

$$\widehat{\boldsymbol{Z}} = \arg\min_{\boldsymbol{Z}} \tilde{f}(\boldsymbol{Z}).$$

If either $\alpha \neq 0$ or $\boldsymbol{A}$ has rank m, then the unique global minimizer is

$$\widehat{\boldsymbol{Z}} = (\boldsymbol{\mu}_{\boldsymbol{\xi}}\boldsymbol{\mu}_{\boldsymbol{\xi}}^{\top} + \boldsymbol{I})\boldsymbol{A}^{\top}[\boldsymbol{A}(\boldsymbol{\mu}_{\boldsymbol{\xi}}\boldsymbol{\mu}_{\boldsymbol{\xi}}^{\top} + \boldsymbol{I})\boldsymbol{A}^{\top} + \alpha^2 \boldsymbol{I}_m]^{-1}.$$

where $\alpha = \eta/\beta$.

This result is interesting, but a full-rank (and probably dense) solution $\widehat{\boldsymbol{Z}}$ is impractical for large-scale problems.

A more practical formulation

$$\tilde{f}(\boldsymbol{Z}) = \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\boldsymbol{\mu}_{\boldsymbol{\xi}}\|_2^2 + \|(\boldsymbol{Z}\boldsymbol{A} - \boldsymbol{I}_n)\|_{\mathrm{F}} + \alpha^2 \|\boldsymbol{Z}\|_{\mathrm{F}}^2 .$$

What happens if we add the assumption that $\boldsymbol{\mu}_{\boldsymbol{\xi}} = \mathbf{0}$, and we seek a matrix $\boldsymbol{Z}$ *of rank at most* k?

We now have the problem solved by our inverse approximation theorem:

If $\sigma_k > \sigma_{k+1}$, the unique global minimizer $\widehat{\boldsymbol{Z}} \in \mathbb{R}^{n \times m}$ is

$$\widehat{\boldsymbol{Z}} = \boldsymbol{V}_k \boldsymbol{\Psi}_k \boldsymbol{U}_k^{\top},$$

where $\boldsymbol{V}_k$ contains the first k columns of $\boldsymbol{V}$, $\boldsymbol{U}_k$ contains the first k columns of $\boldsymbol{U}$, and

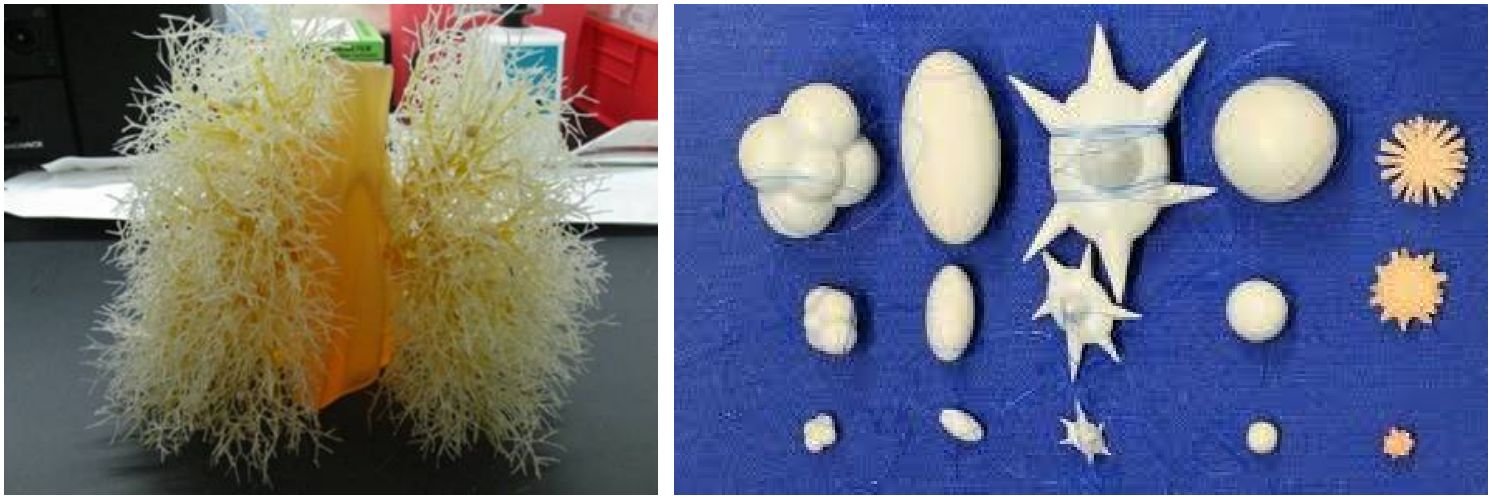
$$\boldsymbol{\Psi}_k = \operatorname{diag}\left(\frac{\sigma_1}{\sigma_1^2 + \alpha^2}, \ldots, \frac{\sigma_k}{\sigma_k^2 + \alpha^2} \right).$$

For general covariance matrices, the solution is similar but written using a generalized SVD.

Experiment 3: A deconvolution example

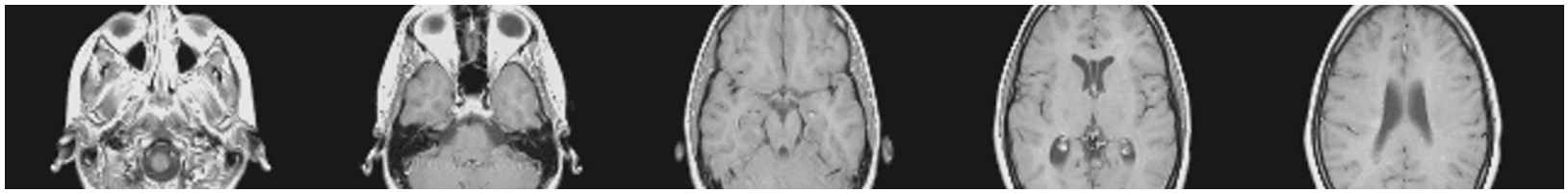
In many imaging applications, calibration data is available.

For example, an imaging device might record an image of a phantom with known properties:

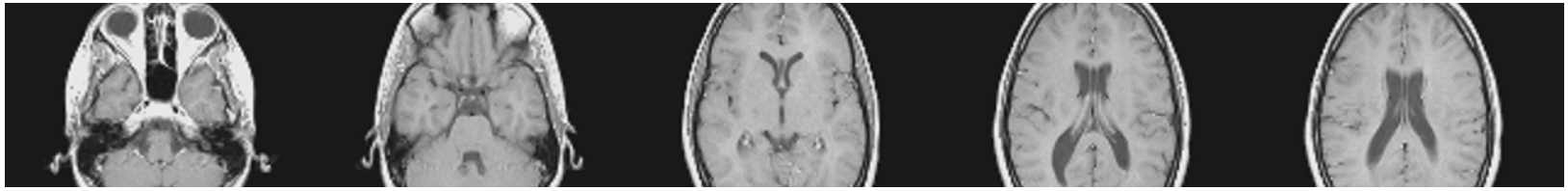


`http://www.fda.gov/`

Our calibration data

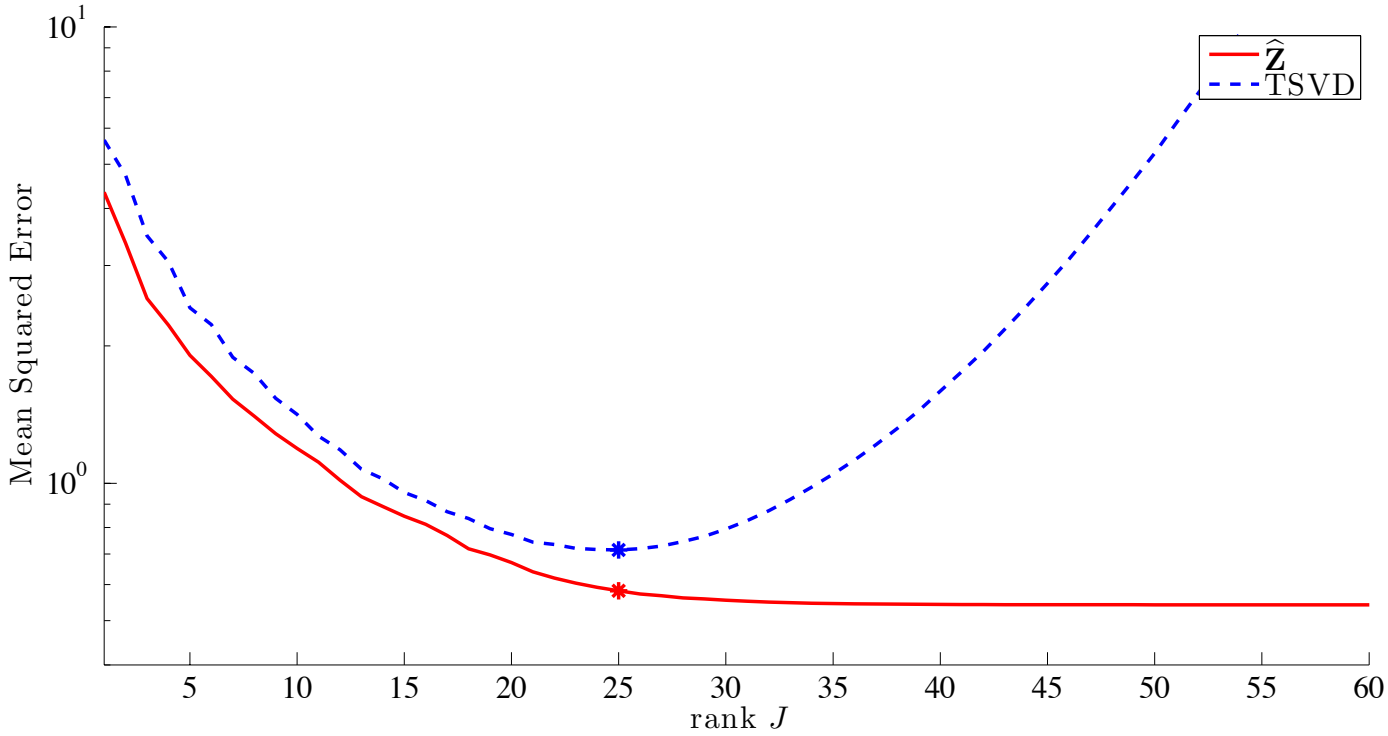


- We use columns from images above to compute a sample covariance matrix.
- We construct our approximate inverse.
- We use columns from images below to evaluate how well it behaves.

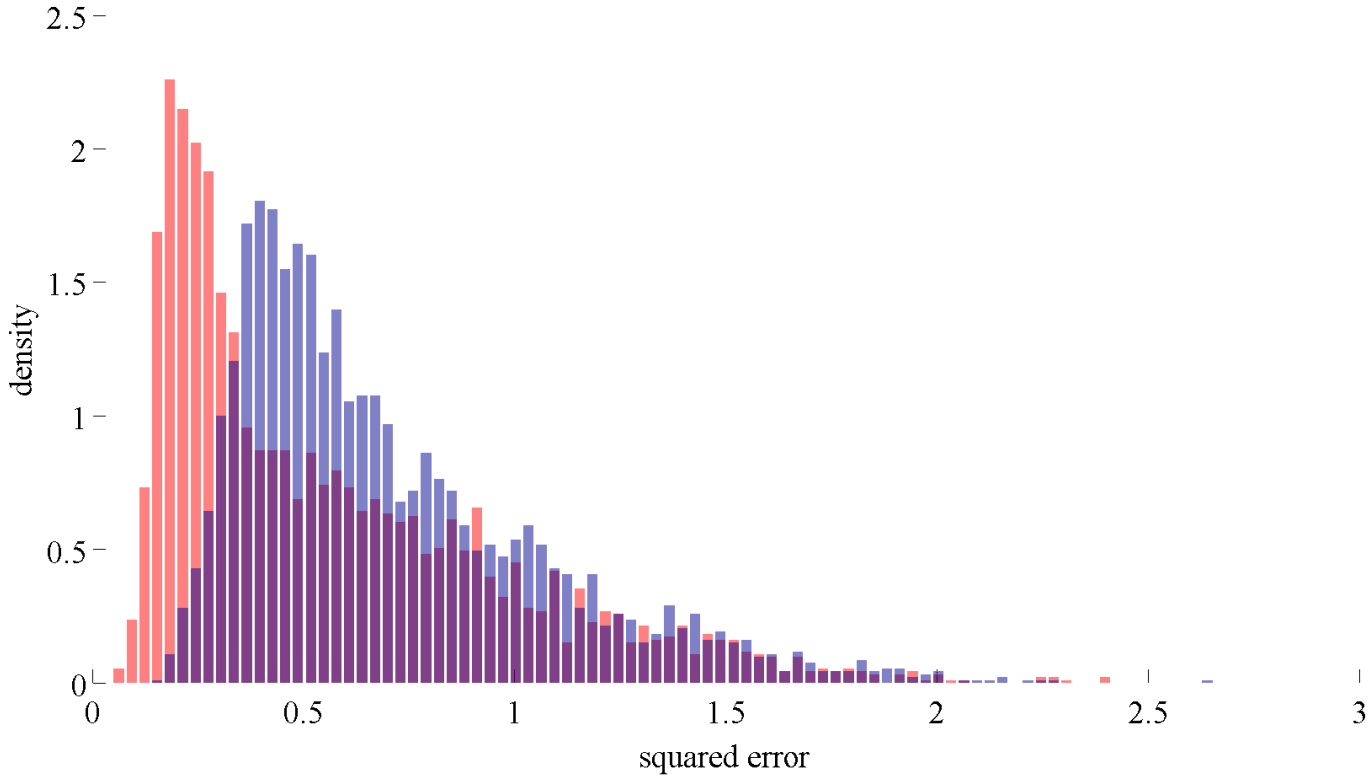


The experiment:

- 1280 possible signals, each 150×1.
- Compute the sample covariance matrix.
- Scale its diagonal by 1.001 to ensure positive definiteness.
- Matrix $\boldsymbol{A} \in \mathbb{R}^{150\times 150}$ represents a Gaussian convolution kernel with variance 4.
- For $\alpha = 0.1$, we compute the optimal low-rank inverse approximation $\widehat{\boldsymbol{Z}}$ for various ranks k.
- Construct 3072 signals, $\boldsymbol{\xi}^{(k)}$, by convolving columns of the (b) images with the Gaussian kernel, and adding noise $\boldsymbol{\delta}^{(k)}$ (normal distribution with covariance matrix $\alpha^2\, \boldsymbol{I}_m$).
- For various ranks k, calculate the mean squared error $\frac{1}{K}\sum_{k=1}^{K} e^{(k)}$, where $e^{(k)} = \left\| \boldsymbol{Z}\boldsymbol{b}^{(k)} - \boldsymbol{\xi}^{(k)} \right\|_2^2$ and $\boldsymbol{b}^{(k)} = \boldsymbol{A}\boldsymbol{\xi}^{(k)} + \boldsymbol{\delta}^{(k)}$, for both $\widehat{\boldsymbol{Z}}$ and the TSVD reconstruction matrix, $\boldsymbol{A}_k^{\dagger}$.



Comparison of mean squared errors for reconstructions using the optimal approximate inverse matrix $\widehat{\boldsymbol{Z}}$ and the TSVD reconstruction matrix $\boldsymbol{A}_k^\dagger$ for various ranks.



Histogram comparison of squared errors for $\widehat{\boldsymbol{Z}}$ (red bars) and TSVD (blue bars) for rank $k = 25$.

Summary: so far

- We can calculate low-rank approximations to matrix (pseudo)inverses.
- These approximations give us useful reconstructions of images blurred by measurement.

Interval approximation

We have discussed some great ways to approximate matrices...

... very useful for the particular situations above.

But sometimes noise is quite abnormal.

Often our data matrix $\boldsymbol{A}$ stores counts.

We might count more or less than the true value, but we can never get a negative count.

An alternative framework

What if we have an uncertainty interval for each matrix element? We would be given matrices $\boldsymbol{L}$ and $\boldsymbol{U}$ satisfying

$$\boldsymbol{L} \leq \boldsymbol{A}_{true} \leq \boldsymbol{U},$$

where $\boldsymbol{A}_{true}$ is unknown.

- We can explicitly impose a nonnegativity assumption, choosing $\boldsymbol{L} \geq \boldsymbol{0}$.
- We can easily accommodate different uncertainties in different matrix elements.
- We can even accommodate missing observations by setting elements of $\boldsymbol{L}$ to zero and elements of $\boldsymbol{U}$ to $+\infty$.
- Our original formulation is a special case, with $\boldsymbol{L} = \boldsymbol{U}$.

Our problem becomes

$$\min_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}} \quad f(\boldsymbol{X}\boldsymbol{Y}-\boldsymbol{Z})$$
$$\ell \leq \boldsymbol{z} \leq \boldsymbol{u},$$

For f, any matrix norm can be used, and we can also include a weight matrix $\boldsymbol{W}$ if we care more about keeping some particular elements within their bounds.

So far, our problem is ill-posed.

If $\boldsymbol{X}\boldsymbol{Y} = \boldsymbol{Z}$ solve the problem,

then (for example) so do $\gamma\boldsymbol{X}, \gamma^{-1}\boldsymbol{Y}, \boldsymbol{Z}$ for any positive γ.

Added constraints

- We add constraints:

$$\boldsymbol{X} \geq \mathbf{0} \qquad \text{and} \qquad \boldsymbol{Y} \geq \mathbf{0}.$$

  Still ill-posed.

- It is also useful to add terms to the minimization function to "encourage" sparsity in the factors:

$$\boldsymbol{e}^T \boldsymbol{x} + \boldsymbol{e}^T \boldsymbol{y}.$$

  Now the γ non-uniqueness goes away.

  BUT the sparsest $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the zero matrices, so we can't weight these terms too heavily without risking a rank-deficient solution to our problem.

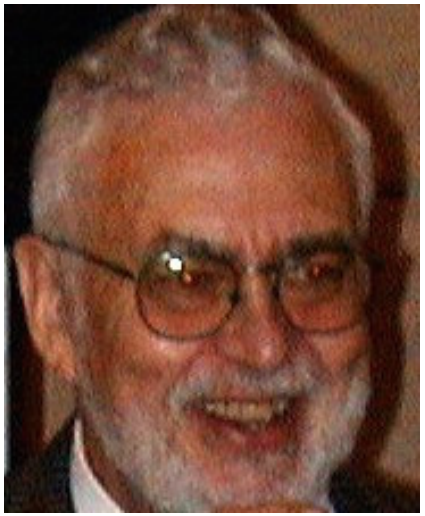
  So to balance these terms, we also can include these:

$$-\log\det(\boldsymbol{X}^T\boldsymbol{X}) - \log\det(\boldsymbol{Y}\boldsymbol{Y}^T).$$

$\log\det$ is scary!

$$\hat{F}(\boldsymbol{X}) = -\log\det(\boldsymbol{X}^T\boldsymbol{X}),$$

- Both $\boldsymbol{X}^T\boldsymbol{X}$ and $\boldsymbol{Y}\boldsymbol{Y}^T$ are small (5×5 or 50×50 in our examples, so evaluation is inexpensive.
- The partial derivatives of $\hat{F}(\boldsymbol{X})$, arranged as a matrix, are $2\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}$, so $\hat{F}(\boldsymbol{X})$ and its gradient are easily calculated using (compact) QR factorization of $\boldsymbol{X}$.
- The Hessian matrix with respect to entries in each row of $\boldsymbol{X}$ is also easy to calculate given the (compact) QR factors.



$\boldsymbol{Z}$ is scary!

Up until now, we have had only $k(m+n)$ variables.

But $\boldsymbol{Z}$ adds mn more!!

$\boldsymbol{Z}$ is scary!

Really, really scary!

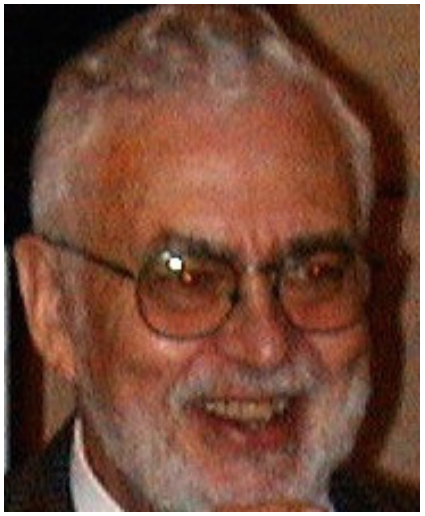
Up until now, we have had only $k(m+n)$ variables.

But $\boldsymbol{Z}$ adds mn more!!

Fortunately, I can tell you an optimal choice of $\boldsymbol{Z}$ for any given $\boldsymbol{X}$ and $\boldsymbol{Y}$:

$$z_{ij}(\boldsymbol{X}, \boldsymbol{Y}) = \begin{cases} \ell_{ij}, & (\boldsymbol{XY})_{ij} \leq \ell_{ij} \\ (\boldsymbol{XY})_{ij}, & \ell_{ij} \leq (\boldsymbol{XY})_{ij} \leq u_{ij}, \\ u_{ij}, & u_{ij} \leq (\boldsymbol{XY})_{ij}. \end{cases}$$

So algorithms that alternate updating $\boldsymbol{X}$ and updating $\boldsymbol{Y}$ can be used, and we update $\boldsymbol{Z}$ using this formula.



Alternating algorithms

$$\min_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}} \|\boldsymbol{X}\boldsymbol{Y}-\boldsymbol{Z}\|_F^2+\alpha_1 \boldsymbol{e}^T\boldsymbol{x}+\alpha_2 \boldsymbol{e}^T\boldsymbol{y}-\alpha_3(\log\det(\boldsymbol{X}^T\boldsymbol{X})-\log\det(\boldsymbol{Y}\boldsymbol{Y}^T))$$

Repeat

Update $\boldsymbol{X}$.
Determine the optimal $\boldsymbol{Z}$ based on the current $\boldsymbol{X}$ and $\boldsymbol{Y}$.
Update $\boldsymbol{Y}$.
Determine the optimal $\boldsymbol{Z}$ based on the current $\boldsymbol{X}$ and $\boldsymbol{Y}$.

Until convergence.

- Blue steps: not seen in previous algorithms, because now $\boldsymbol{L} \neq \boldsymbol{U}$.
- Red steps: various choices in the literature.

Updating $\boldsymbol{X}$, Option 1

Relevant terms:

$$h(\boldsymbol{X}) = \|\boldsymbol{X}\boldsymbol{Y} - \boldsymbol{Z}\|_F^2 + \alpha_1 \boldsymbol{e}^T \boldsymbol{x} - \alpha_3 \log\det(\boldsymbol{X}^T \boldsymbol{X})$$

Option 1 Advocated by Lee & Seung (but without $\log\det$):
Minimize $h(\boldsymbol{X})$ (or at least take a step that reduces $h(\boldsymbol{X})$) and then set negative entries in $\boldsymbol{X}$ to zero.

- Advantages: The computation is inexpensive!
- Disadvantages: No guarantee that the new $\boldsymbol{X}$ reduces $h(\boldsymbol{X})$.

Updating $\boldsymbol{X}$, Option 2

Relevant terms:

$$h(\boldsymbol{X}) = \|\boldsymbol{X}\boldsymbol{Y} - \boldsymbol{Z}\|_F^2 + \alpha_1 \boldsymbol{e}^T \boldsymbol{x} - \alpha_3 \log\det(\boldsymbol{X}^T \boldsymbol{X})$$

Option 2 Advocated by Kim, Sra, & Dhillon (but without $\log\det$):
Minimize $h(\boldsymbol{X})$ (or at least take a step that reduces $h(\boldsymbol{X})$) while maintaining nonnegativity of $\boldsymbol{X}$.

- Advantages: The new $\boldsymbol{X}$ reduces $h(\boldsymbol{X})$ and maintains nonnegativity.
- Advantages: Convergence proof for the alternating algorithm, but under an obnoxious assumption: $\boldsymbol{X}$ and $\boldsymbol{Y}$ never become rank deficient. Unfortunately, in practice, they do (without the red term)!
- Disadvantages: The computation is expensive!

Notice that (without the red term) each row of $\boldsymbol{X}$ can be updated independently.

New option: Downhill, constraint satisfying, and full rank

New option: $h(\boldsymbol{X}) = \|\boldsymbol{X}\boldsymbol{Y} - \boldsymbol{Z}\|_F^2 + \alpha_1 \boldsymbol{e}^T \boldsymbol{x} - \alpha_3 \log\det(\boldsymbol{X}^T \boldsymbol{X})$

- Search direction: $\boldsymbol{s} = -\boldsymbol{P}\boldsymbol{D}\boldsymbol{P}\boldsymbol{g}$, where
  - $\boldsymbol{g}$ is the gradient of h with respect to $\boldsymbol{x}$,
  - $\boldsymbol{D}$ is a positive definite scaling matrix,
  - $\boldsymbol{P}$ is a projection matrix that sets $(\boldsymbol{P}\boldsymbol{t})_j$ to zero if $x_j = 0$ and $t_j > 0$.
- Update $\boldsymbol{x} \leftarrow \boldsymbol{x} + \nu \boldsymbol{s}$, where ν is determined by a line search.

This is the same search direction used by Kim, Sra, & Dhillon, except that we include the $\log\det$ term in our problem formulation.

So now we retain full rank in the iterates.

What can we prove?

Assume that we never take an uphill step. So

$$F(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) = \|\boldsymbol{X}\boldsymbol{Y} - \boldsymbol{Z}\|_F^2 + \alpha_1 \boldsymbol{e}^T \boldsymbol{x} + \alpha_2 \boldsymbol{e}^T \boldsymbol{y} - \alpha_3(\log\det(\boldsymbol{X}^T\boldsymbol{X}) + \log\det(\boldsymbol{Y}\boldsymbol{Y}^T))$$

is never larger than its initial value f_0.

- The $\log\det$ term is (usually) small.
- The linear term gives us a bound on $\|\boldsymbol{Y}\|$.
- The $\log\det$ term then gives us a nonzero bound on the smallest eigenvalue of $\boldsymbol{Y}\boldsymbol{Y}^T$.
- So we can guarantee that our iterates are full rank!

$$F(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) = \|\boldsymbol{X}\boldsymbol{Y} - \boldsymbol{Z}\|_F^2 + \alpha_1 \boldsymbol{e}^T \boldsymbol{x} + \alpha_2 \boldsymbol{e}^T \boldsymbol{y} - \alpha_3 (\log\det(\boldsymbol{X}^T \boldsymbol{X}) + \log\det(\boldsymbol{Y}\boldsymbol{Y}^T))$$

Notice:

- We have guaranteed that $(\boldsymbol{Y}\boldsymbol{Y}^T)^{-1}$ is uniformly bounded throughout our iteration.
- $\boldsymbol{Y}\boldsymbol{Y}^T$ is the Hessian matrix w.r.t. $\boldsymbol{X}$ variables for the quadratic terms.
- Linear systems involving $\boldsymbol{Y}\boldsymbol{Y}^T$ are easy to solve using a Cholesky factorization, or a QR factorization of $\boldsymbol{Y}^T$, and we already needed a factorization in order to evaluate the $\log\det$ term.

Therefore, $(\boldsymbol{Y}\boldsymbol{Y}^T)^{-1}$ is an ideal candidate for the scaling matrix $\boldsymbol{D}$ in the $\boldsymbol{X}$ iteration and makes the iteration Newton-like.

Interchange $\boldsymbol{X}$ and $\boldsymbol{Y}$ in this discussion to get the same result for the $\boldsymbol{Y}$ iteration.

Results: Document classification

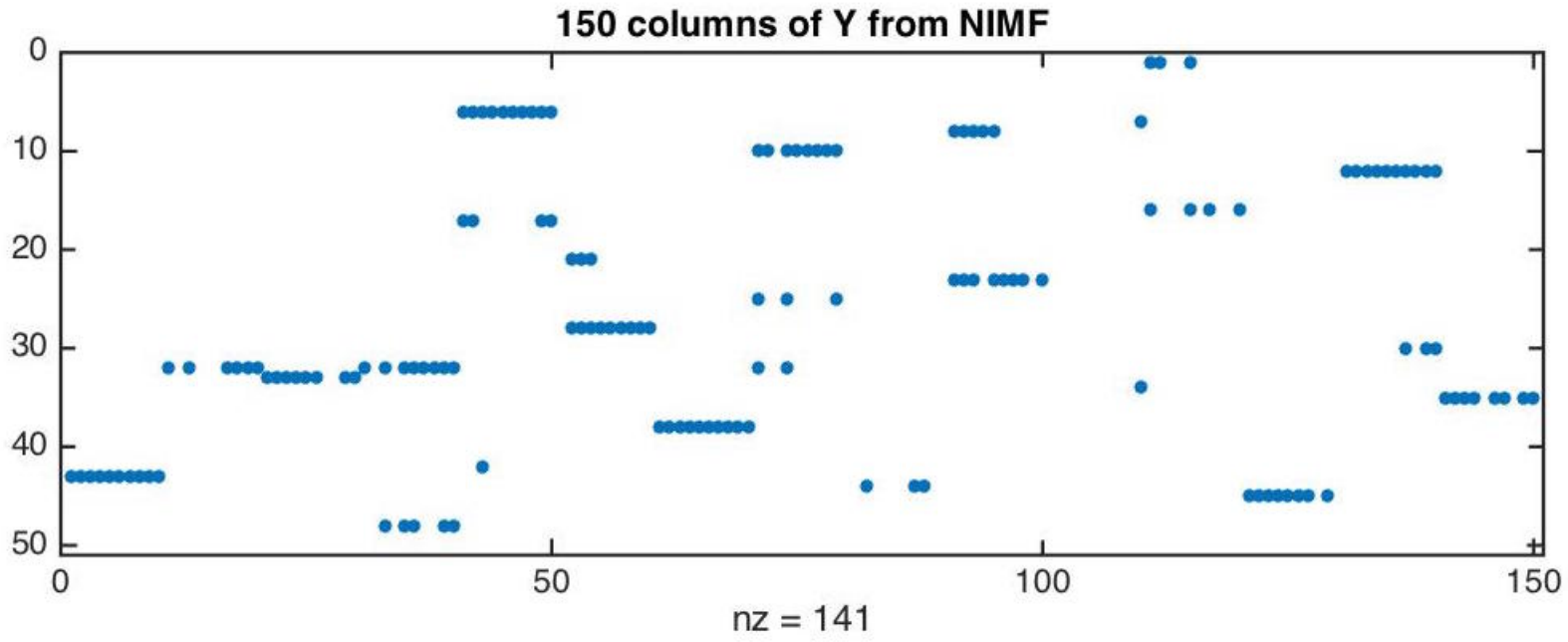
DUC 2004 data: Term-document matrix for 500 documents (some repeats) that are "correctly" classified in 50 classes of 10 documents each.

Measures of correctness: Let
A = number of agreements: document i and j in same/different class
D = number of disagreements,
so that A+D = comb(n,2). Then

.9846 = RI = Rand index (1971) = A / comb(n,2) = prob agree
.9692 = HI = Hubert index (1977) = (A - D) / comb(n,2) = RI - MI
.6252 = AR = adjusted RI, corrected for chance Hubert+Arabie (1985)
.7500 = AMI = adjusted mutual index

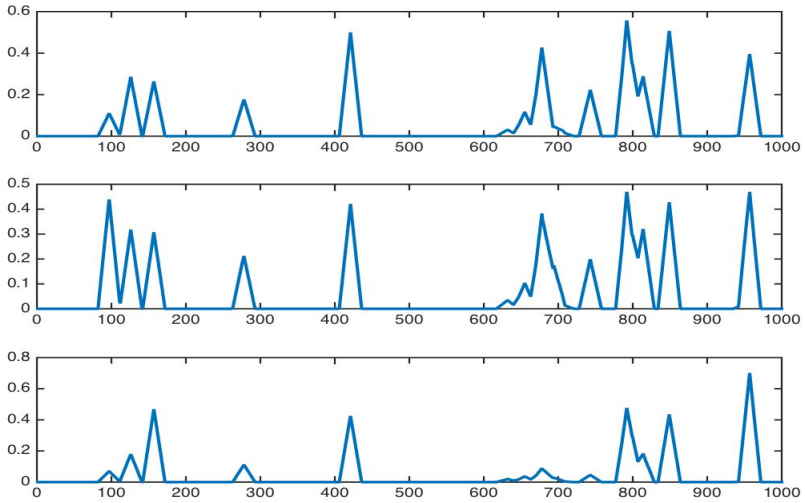
Comparable to competing methods.

The Computed $\boldsymbol{Y}$ matrix



Results: Spectroscopy

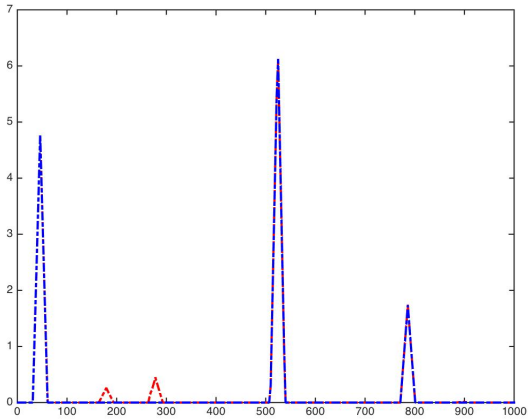
Given 50 noisy combinations of the 5 underlying spectra

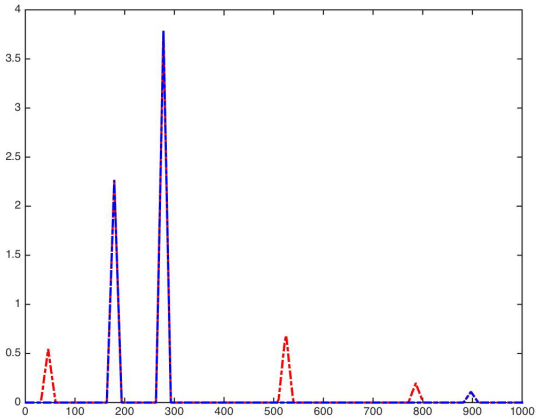


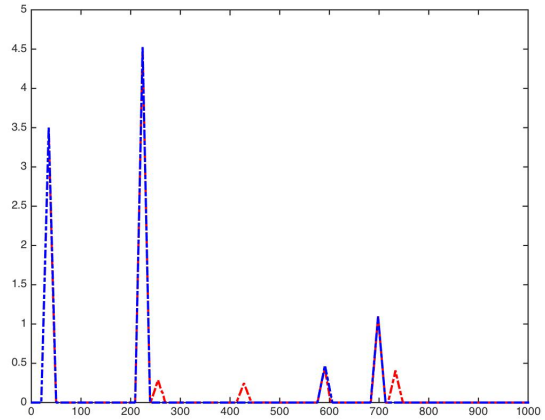
(and 47 more)

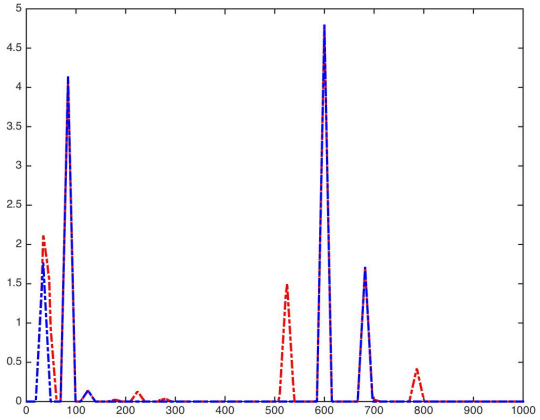
Reconstruct the 5 spectra.

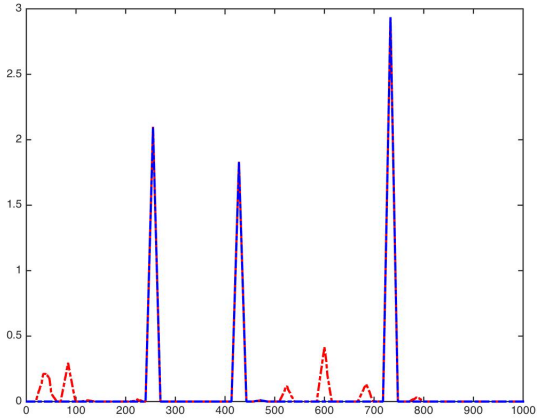
Blue = True, Red = Computed











Conclusions

A fresh look at low-rank matrix approximation is productive and useful!

- Inverse Approximation of Matrices
  - We can calculate low-rank approximations to matrix (pseudo)inverses.
  - These approximations give us useful reconstructions of images blurred by measurement when calibration data is available.
- Approximation of Interval Matrices
  - We have a new formulation of matrix approximation that allows error bounds and weights on each matrix entry.
  - We have a descent algorithm that takes Newton-like steps to improve the free variables.
  - We have demonstrated promising results for document classification and signal recovery.

References for Inverse Approximation:

- Julianne Chung, Matthias Chung, and Dianne P. O'Leary "Optimal Regularized Low-Rank Inverse Approximation," Linear Algebra and Its Applications, 468(1) (2015) 260-269.
- Julianne M. Chung, Matthias Chung, and Dianne P. O'Leary, "Optimal Filters from Calibration Data for Image Deconvolution with Data Acquisition Errors," Journal of Mathematical Imaging and Vision, 44(3), pp. 366–374 (2012)

Thank you, Gene!

And thank you to my collaborators

Julianne Chung

Matthias Chung

John Conroy

Yi-Kai Liu

www.cs.umd.edu/users/oleary/talkview.pdf