SVD Analysis

Per Christian Hansen

DTU Compute
Department of Applied Mathematics and Computer Science
Technical University of Denmark

Vectors and Images/Sinograms

Note that in the discretized problem $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$, the image and the sinogram are represented by the *vectors* $\boldsymbol{x}$ and $\boldsymbol{b}$, respectively.

While this is a convenient notation when we need the language of linear algebra, they are really 2D arrays and they should be visualized as such.

In Matlab notation:

```
imagesc( reshape(x,N,N) ), axis image
imagesc( reshape(b,N_s,N_θ) ), axis image
```

where N_s = number of detector pixels, and N_θ = number of projections.

Going from an image `X` to a vector $\boldsymbol{x}$ is simple: just write $\boldsymbol{x}$ `= X(:)`.

Matrix Notation and Interpretation

All vectors are *column vectors*. For the system matrix we have

$$\boldsymbol{A} = \begin{pmatrix} | & | & & | \\ \boldsymbol{c}_1 & \boldsymbol{c}_2 & \cdots & \boldsymbol{c}_n \\ | & | & & | \end{pmatrix} = \begin{pmatrix} \text{---} & \boldsymbol{r}_1^T & \text{---} \\ & \vdots & \\ \text{---} & \boldsymbol{r}_m^T & \text{---} \end{pmatrix} .$$

The matrix $\boldsymbol{A}$ maps the discretized absorption coefficients (the vector $\boldsymbol{x}$) to the data in the detector pixels (the elements of the vector $\boldsymbol{b}$) via:

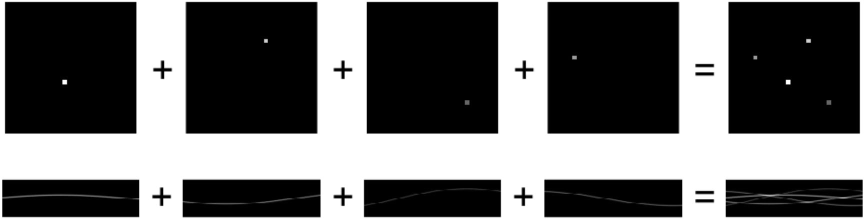
$$\boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \boldsymbol{A}\,\boldsymbol{x} = \underbrace{x_1\,\boldsymbol{c}_1 + x_2\,\boldsymbol{c}_2 + \cdots + x_n\,\boldsymbol{c}_n}_{\text{linear combination of columns}} = \begin{pmatrix} \boldsymbol{r}_1^T\boldsymbol{x} \\ \boldsymbol{r}_2^T\boldsymbol{x} \\ \vdots \\ \boldsymbol{r}_m^T\boldsymbol{x} \end{pmatrix} .$$

See next slides for examples of the column and row interpretations.

Example of Column Interpretation

A 32×32 image has four nonzero pixels with intensities 1, 0.8, 0.6, 0.4. In the vector $\boldsymbol{x}$, these four pixels correspond to entries 468, 618, 206, 793. Hence the sinogram, represented as a vector $\boldsymbol{b}$, takes the form

$$\boldsymbol{b} = 0.6\, \boldsymbol{c}_{206} + 1.0\, \boldsymbol{c}_{468} + 0.8\, \boldsymbol{c}_{618} + 0.4\, \boldsymbol{c}_{793}.$$



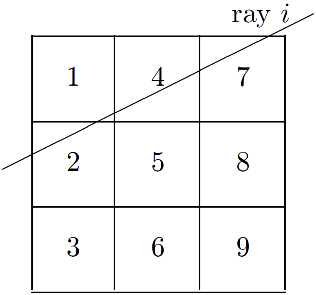
Note that each pixel is mapped to a single sinusoidal curve in the sinogram.

Example of Row Interpretation

The ith row of $\boldsymbol{A}$ maps $\boldsymbol{x}$ to detector element i via the ith ray:

$$b_i = \boldsymbol{r}_i^T \boldsymbol{x} = \sum_{j=1}^{n} a_{ij}\, x_j, \qquad i = 1, 2, \ldots, m.$$

This inner product approximates the line integral along ray i in the Radon transform.



A small example:

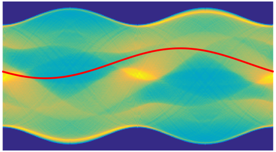
$$\begin{aligned} a_{ij} &= \text{length of ray } i \text{ in pixel } j \\ \boldsymbol{r}_i &= [\; a_{i1} \;\; a_{i2} \;\; 0 \;\; a_{i4} \;\; 0 \;\; 0 \;\; a_{i7} \;\; 0 \;\; 0 \;] \\ b_i &= \boldsymbol{r}_i^T \boldsymbol{x} = a_{i1}x + a_{i2}y + a_{i4}x_4 + a_{i7}x_7 \end{aligned}$$

Back Projection and the Matrix Transpose

Recall the back projection,

$$\mathcal{R}^\sharp[g](x_1, x_2) = \int_0^{2\pi} g(x_1 \cos\theta + x_2 \sin\theta, \theta)\, d\theta,$$

where we integrate g along a sinusoidal curve in the sinogram:



Multiplication with the matrix transpose performs this operation:

$$\boldsymbol{A}^T \boldsymbol{b} = \begin{pmatrix} | & & | \\ \boldsymbol{c}_1 & \cdots & \boldsymbol{c}_n \\ | & & | \end{pmatrix}^T \boldsymbol{b} = \begin{pmatrix} \text{---} & \boldsymbol{c}_1^T & \text{---} \\ & \vdots & \\ \text{---} & \boldsymbol{c}_n^T & \text{---} \end{pmatrix} \boldsymbol{b} = \begin{pmatrix} \boldsymbol{c}_1^T \boldsymbol{b} \\ \vdots \\ \boldsymbol{c}_n^T \boldsymbol{b} \end{pmatrix}$$

where each inner product $\boldsymbol{c}_j^T \boldsymbol{b}$ corresponds to the above integration.

And Now: The Singular Value (SVD)

Assume that $\boldsymbol{A}$ is $m \times n$ and, for simplicity, that $m \geq n$:

$$\boldsymbol{A} = \boldsymbol{U}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^T = \sum_{i=1}^{n} \boldsymbol{u}_i\,\sigma_i\,\boldsymbol{v}_i^T .$$

Here, $\boldsymbol{\Sigma}$ is a diagonal matrix with the *singular values*, satisfying

$$\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)\ , \qquad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0\ .$$

The matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ consist of *singular vectors*

$$\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n)\ , \qquad \boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$$

and both matrices have orthonormal columns: $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}_n$.

Then $\|\boldsymbol{A}\|_2 = \sigma_1$, $\|\boldsymbol{A}^{-1}\|_2 = \|\boldsymbol{V}\,\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T\|_2 = \sigma_n^{-1}$, and

$$\mathrm{cond}(\boldsymbol{A}) = \|\boldsymbol{A}\|_2\,\|\boldsymbol{A}^{-1}\|_2 = \sigma_1/\sigma_n.$$

Important SVD Relations

Relations similar to the analysis of the Radon transform:

$$\boldsymbol{A}\,\boldsymbol{v}_i = \sigma_i\,\boldsymbol{u}_i, \qquad \|\boldsymbol{A}\,\boldsymbol{v}_i\|_2 = \sigma_i, \qquad i = 1,\ldots,n.$$

Also, if $\boldsymbol{A}$ is nonsingular, then

$$\boldsymbol{A}^{-1}\boldsymbol{u}_i = \sigma_i^{-1}\,\boldsymbol{v}_i, \qquad \|\boldsymbol{A}^{-1}\boldsymbol{u}_i\|_2 = \sigma_i^{-1}, \qquad i = 1,\ldots,n.$$

These equations are related to the solution:

$$\begin{aligned}
\boldsymbol{x} &= \sum_{i=1}^{n} (\boldsymbol{v}_i^T\boldsymbol{x})\,\boldsymbol{v}_i \\
\boldsymbol{A}\,\boldsymbol{x} &= \sum_{i=1}^{n} \sigma_i\,(\boldsymbol{v}_i^T\boldsymbol{x})\,\boldsymbol{u}_i\ , \qquad \boldsymbol{b} = \sum_{i=1}^{n} (\boldsymbol{u}_i^T\boldsymbol{b})\,\boldsymbol{u}_i \\
\boldsymbol{A}^{-1}\boldsymbol{b} &= \sum_{i=1}^{n} \frac{\boldsymbol{u}_i^T\boldsymbol{b}}{\sigma_i}\,\boldsymbol{v}_i\ .
\end{aligned}$$

The Naive or "Plain Vanilla Solution"

From now on, the coefficient matrix $\boldsymbol{A}$ is allowed to have more rows than columns, i.e.,

$$\boldsymbol{A} \in \mathbb{R}^{m \times n} \qquad \text{with} \qquad m \geq n.$$

For $m > n$ it is natural to consider the least squares problem

$$\min_{\boldsymbol{x}} \|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2.$$

Then we refer to "the naive solution" $\boldsymbol{x}^{\text{naive}}$ as either the solution $\boldsymbol{A}^{-1}\boldsymbol{b}$ (when $m = n$) or the least squares solution (when $m > n$). While these solutions are straightforward to compute, one should not be naive and assume that they are always useful in CT.

We emphasize the convenient fact that both solutions has precisely the same SVD expansion in both cases:

$$\boldsymbol{x}^{\text{naive}} = \sum_{i=1}^{n} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i}\,\boldsymbol{v}_i.$$

Computing the SVD in MATLAB

The matrix $\boldsymbol{A}^T\boldsymbol{A}$ is symmetric and hence 1) its eigenvalues are real, and 2) its eigenvectors are real and orthonormal (standard linear algebra stuff):

$$\boldsymbol{A}^T\boldsymbol{A}\,\boldsymbol{v}_i = \sigma_i^2\,\boldsymbol{v}_i, \qquad i = 1, 2, \ldots, n.$$

I.e., the right singular vectors $\boldsymbol{v}_i$ are the eigenvectors of $\boldsymbol{A}^T\boldsymbol{A}$ and the squared singular values are the corresponding eigenvalues.

This is **not** how the SVD should be computed, due to the bad influence of rounding errors. Use only good numerical software. In MATLAB:

- Use `[U,S,V] = svd(A)` or `[U,S,V] = svd(A,0)` to compute the full or "economy-size" SVD.
- If `A` is sparse the use `svd(full(A))` or `svd(full(A),0)`.
- Use `[U,S,V] = svds(A)` to efficiently compute a partial SVD (the largest singular values and corresponding singular vectors).

What the SVD Looks Like – A Simple 1D Example

We consider a very simple problem (not related to CT): determine the elements of the vector $\boldsymbol{x} \in \mathbb{R}^n$ from its cumulative sums:

$$b_i = \frac{1}{n} \sum_{j=1}^{i} x_j \ , \qquad i = 1, 2, \ldots, n \ . \tag{1}$$

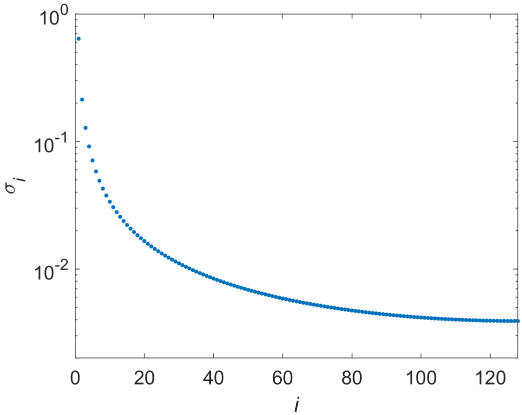
The weighting $1/n$ is for convenience.

Let b_i be elements of a vector $\boldsymbol{b}$. Then we have the relation $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \frac{1}{n} \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ 1 & 1 & 0 & 0 & \cdots \\ 1 & 1 & 1 & 0 & \cdots \\ 1 & 1 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} , \qquad \boldsymbol{A} \in \mathbb{R}^{n \times n} \ . \tag{2}$$

The solution to $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$ is quite sensitive to errors in the right-hand side.

The Singular Values

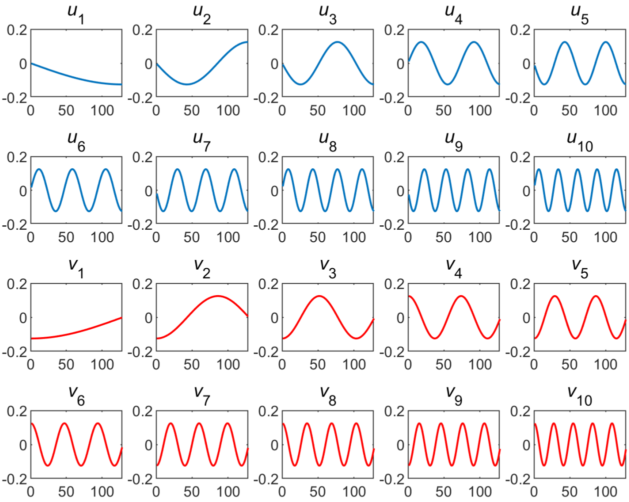


The singular values of $\boldsymbol{A}$ decay, according to their definition.

If we increase the problem szie n they slowly approach zero.

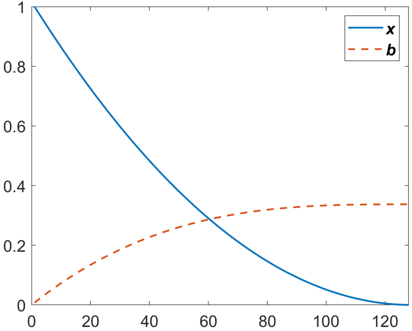
This is similar to the behavior of the singular values of the Radon transform.

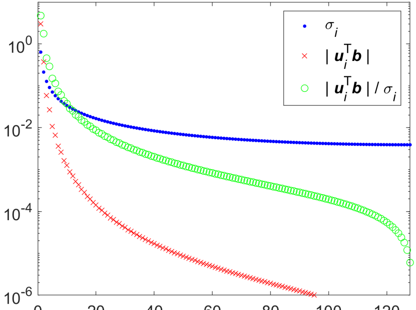
The First 10 Singular Vectors



The number of oscillations increases as the index i increases. Again, this is similar to the behavior of the singular functions of the Radon transform.

SVD Analysis of Cum-Sum Problem – No Noise in the Data





Left: the solution $\boldsymbol{x}$ and the corresponding noise-free right-hand side $\boldsymbol{b}$.

Right: ingredients of the naive solution.

The singular values decay rather slowly and that the right-hand side's coefficients $|\boldsymbol{u}_i^T\boldsymbol{b}|$ decay faster than the singular values σ_i.

Ill-Conditioned Problems

Discrete ill-posed problems are characterized by having coefficient matrices with a large condition number. The solution is very sensitive to errors in the data.

Specifically, assume that the exact and perturbed solutions $\bar{\boldsymbol{x}}$ and $\boldsymbol{x}$ satisfy

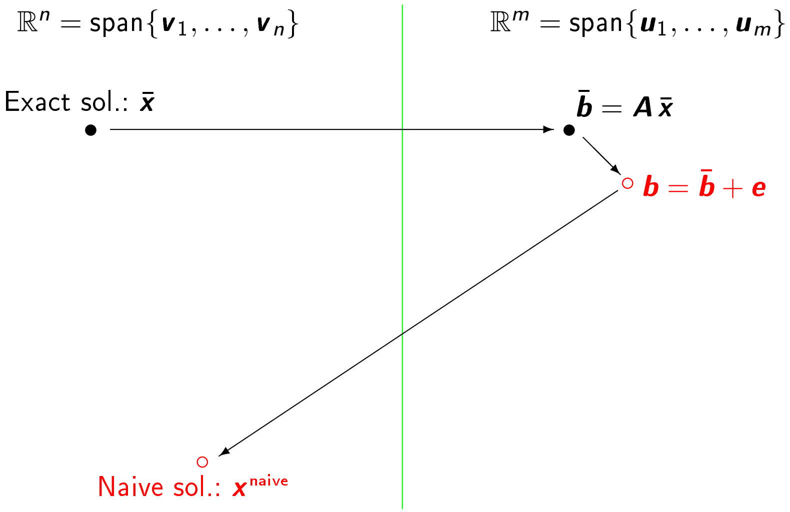
$$\boldsymbol{A}\,\bar{\boldsymbol{x}} = \bar{\boldsymbol{b}}, \qquad \boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b} = \bar{\boldsymbol{b}} + \boldsymbol{e},$$

where $\boldsymbol{e}$ denotes the perturbation (the errors and noise). Then classical perturbation theory leads to the bound

$$\frac{\|\bar{\boldsymbol{x}} - \boldsymbol{x}\|_2}{\|\bar{\boldsymbol{x}}\|_2} \leq \mathrm{cond}(\boldsymbol{A})\,\frac{\|\boldsymbol{e}\|_2}{\|\bar{\boldsymbol{b}}\|_2}.$$

Since $\mathrm{cond}(\boldsymbol{A}) = \sigma_1/\sigma_n$ is large, this implies that $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ can be very far from $\bar{\boldsymbol{x}}$.

The Geometry of Ill-Conditioned Problems



SVD Insight About the Noise

Recall that the naive solution is given by

$$\boldsymbol{x}^{\text{naive}} = \sum_{i=1}^{n} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i .$$

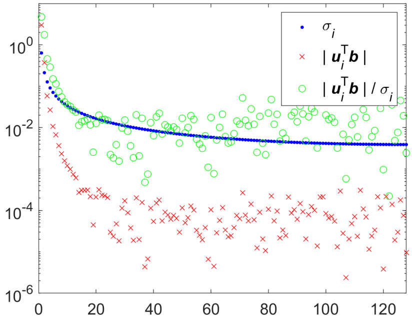
When noise is present in the data $\boldsymbol{b} = \bar{\boldsymbol{b}} + \boldsymbol{e}$, then

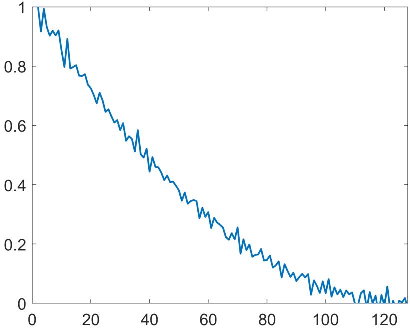
$$\boldsymbol{u}_i^T \boldsymbol{b} = \boldsymbol{u}_i^T \bar{\boldsymbol{b}} + \boldsymbol{u}_i^T \boldsymbol{e} \approx \begin{cases} \boldsymbol{u}_i^T \bar{\boldsymbol{b}}, & |\boldsymbol{u}_i^T \bar{\boldsymbol{b}}| > |\boldsymbol{u}_i^T \boldsymbol{e}| \\ \boldsymbol{u}_i^T \boldsymbol{e}, & |\boldsymbol{u}_i^T \bar{\boldsymbol{b}}| < |\boldsymbol{u}_i^T \boldsymbol{e}|. \end{cases}$$

We note that:

- Due to the Picard condition, the noise-free $|\boldsymbol{u}_i^T \bar{\boldsymbol{b}}|$ decay.
- The "noisy" components $|\boldsymbol{u}_i^T \boldsymbol{b}|$ are those for which $|\boldsymbol{u}_i^T \bar{\boldsymbol{b}}|$ is small,
- and they correspond to the smaller singular values σ_i.

SVD Analysis of Cum-Sum Problem – With Noise in Data





Left: SVD analysis of noisy problem; the SVD coefficients for the noisy right-hand side level off at 10^{-4} = the standard deviation of the noise.

Right: the naive solution dominated by a high-frequency perturbation.

Spectral Filtering

Many of the noise-reducing methods treated in this course produce solutions which can be expressed as a filtered SVD expansion of the form

$$\boldsymbol{x}_{\text{filt}} = \sum_{i=1}^{n} \varphi_i \, \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \, \boldsymbol{v}_i ,$$

where φ_i are the filter factors associated with the method.

These methods are called *spectral filtering methods* because the SVD basis can be considered as a spectral basis.

A simple approach is to discard the SVD coefficients corresponding to the smallest singular values:

$$\varphi_i^{\text{TSVD}} = \begin{cases} 1 & i = 1, 2, \dots, k \\ 0 & \text{else.} \end{cases}$$

More sophisticated methods will be discussed in the third week.

Truncated SVD

Define the truncated SVD (TSVD) solution as

$$\boldsymbol{x}_k = \sum_{i=1}^{n} \varphi_i^{\mathrm{TSVD}} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i = \sum_{i=1}^{k} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i, \qquad k < n.$$

Theorem. Let $\boldsymbol{b} = \bar{\boldsymbol{b}} + \boldsymbol{e}$ and let $\boldsymbol{x}_k$ and $\bar{\boldsymbol{x}}_k$ denote the TSVD solutions computed with the *same* k.

Then

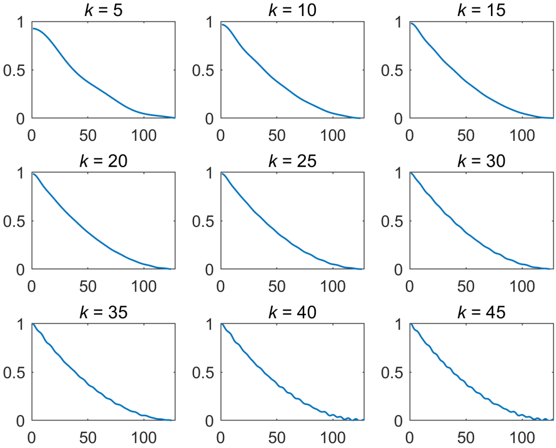
$$\frac{\|\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k\|_2}{\|\boldsymbol{x}_k\|_2} \le \frac{\sigma_1}{\sigma_k} \frac{\|\boldsymbol{e}\|_2}{\|\boldsymbol{A}\,\boldsymbol{x}_k\|_2}.$$

We see that the condition number for the TSVD solution is

$$\kappa_k = \frac{\sigma_1}{\sigma_k}$$

and it can be much smaller than $\mathrm{cond}(\boldsymbol{A}) = \sigma_1/\sigma_n$.

TSVD Solutions $\boldsymbol{x}_k$ to the Noisy Cum-Sum Problem



As we increase the truncation parameter k we include more SVD components and also more noise in $\boldsymbol{x}_k$.

At some point the noise becomes visible and then $\boldsymbol{x}_k$ starts to deteriorate.

The Truncation Parameter

Note: the truncation parameter k in

$$\boldsymbol{x}_k = \sum_{i=1}^{k} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i$$

is dictated by the coefficients $\boldsymbol{u}_i^T \boldsymbol{b}$, not the singular values.

Basically we should choose k as the index i where $|\boldsymbol{u}_i^T \boldsymbol{b}|$ start to "level off" due to the noise.

The TSVD solution and residual norms vary monotonically with k:

$$\|\boldsymbol{x}_k\|_2^2 = \sum_{i=1}^{k} \left(\frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \right) \leq \|\boldsymbol{x}_{k+1}\|_2^2,$$

$$\|\boldsymbol{A}\,\boldsymbol{x}_k - \boldsymbol{b}\|_2^2 = \sum_{i=k+1}^{n} (\boldsymbol{u}_i^T \boldsymbol{b})^2 \geq \|\boldsymbol{A}\,\boldsymbol{x}_{k+1} - \boldsymbol{b}\|_2^2.$$

Where TSVD Fits in the Picture