Separable Nonlinear Least Squares Problems in Image Processing

Julianne Chung and James Nagy
Emory University
Atlanta, GA, USA

Collaborators: Eldad Haber (Emory)
Per Christian Hansen (Tech. Univ. of Denmark)
Dianne O'Leary (University of Maryland)

Inverse Problems in Imaging

Imaging problems are often modeled as:

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{e}$$

where

- $\mathbf{A}$ - large, ill-conditioned matrix
- $\mathbf{b}$ - known, measured (image) data
- $\mathbf{e}$ - noise, statistical properties may be known

Goal: Compute approximation of image $\mathbf{x}$

Inverse Problems in Imaging

A more realistic image formation model is:

$$\mathbf{b} = \mathbf{A}(\mathbf{y})\,\mathbf{x} + \mathbf{e}$$

where

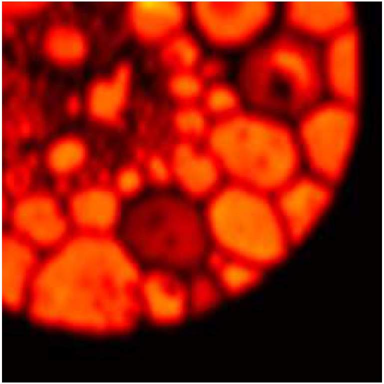
- $\mathbf{A}(\mathbf{y})$ - large, ill-conditioned matrix
- $\mathbf{b}$ - known, measured (image) data
- $\mathbf{e}$ - noise, statistical properties may be known
- $\mathbf{y}$ - parameters defining $\mathbf{A}$, usually approximated

Goal: Compute approximation of image $\mathbf{x}$
and improve estimate of parameters $\mathbf{y}$

Application: Image Deblurring

- $\mathbf{b} = \mathbf{A}(\mathbf{y})\,\mathbf{x} + \mathbf{e}$ = observed image where $\mathbf{y}$ describes blurring function
- Given: $\mathbf{b}$ and an estimate of $\mathbf{y}$
- Standard Image Deblurring: Compute approximation of $\mathbf{x}$
- Better approach: Jointly improve estimate of $\mathbf{y}$ and compute approximation of $\mathbf{x}$.

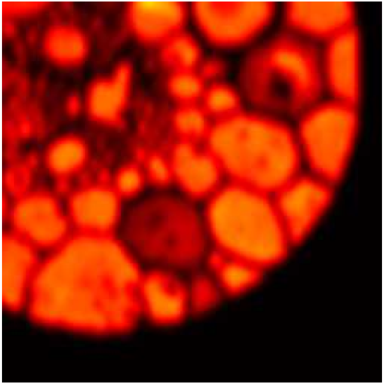
Observed Image



Application: Image Deblurring

- $\mathbf{b} = \mathbf{A}(\mathbf{y})\,\mathbf{x} + \mathbf{e}$ = observed image where $\mathbf{y}$ describes blurring function
- Given: $\mathbf{b}$ and an estimate of $\mathbf{y}$
- Standard Image Deblurring: Compute approximation of $\mathbf{x}$
- Better approach: Jointly improve estimate of $\mathbf{y}$ and compute approximation of $\mathbf{x}$.

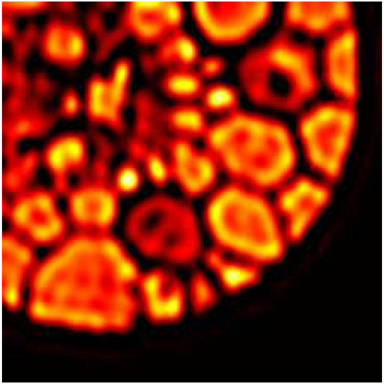
Observed Image



Application: Image Deblurring

- $\mathbf{b} = \mathbf{A}(\mathbf{y})\,\mathbf{x} + \mathbf{e}$ = observed image where $\mathbf{y}$ describes blurring function
- Given: $\mathbf{b}$ and an estimate of $\mathbf{y}$
- Standard Image Deblurring: Compute approximation of $\mathbf{x}$
- Better approach: Jointly improve estimate of $\mathbf{y}$ and compute approximation of $\mathbf{x}$.

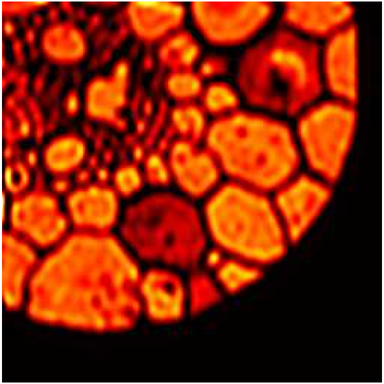
Reconstruction using initial PSF



Application: Image Deblurring

- $\mathbf{b} = \mathbf{A}(\mathbf{y})\,\mathbf{x} + \mathbf{e}$ = observed image where $\mathbf{y}$ describes blurring function
- Given: $\mathbf{b}$ and an estimate of $\mathbf{y}$
- Standard Image Deblurring: Compute approximation of $\mathbf{x}$
- Better approach: Jointly improve estimate of $\mathbf{y}$ and compute approximation of $\mathbf{x}$.

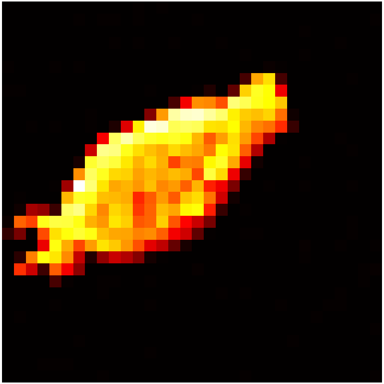
Reconstruction after 8 GN iterations



Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

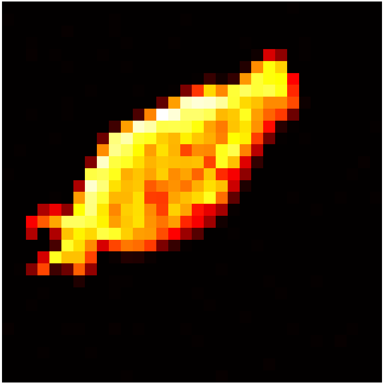
1−th low resolution image



Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

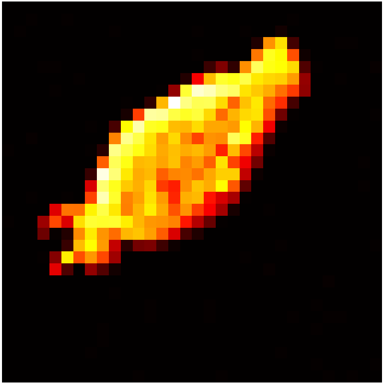
8−th low resolution image



Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

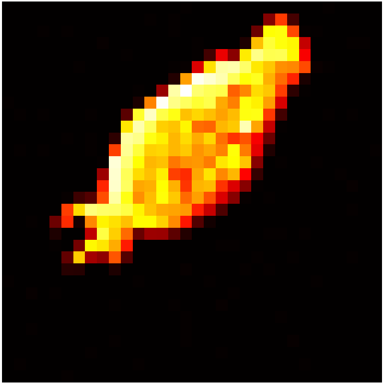
15−th low resolution image



Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

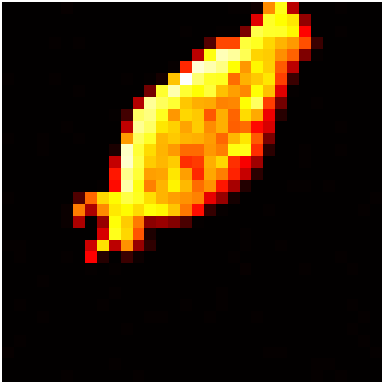
22−th low resolution image



Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

29−th low resolution image



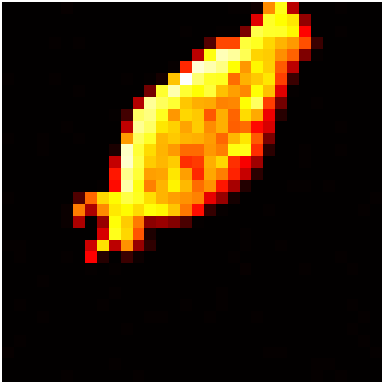
Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

$$\underbrace{\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_m \end{bmatrix}}_{\mathbf{b}} = \underbrace{\begin{bmatrix} \mathbf{A}(\mathbf{y}_1) \\ \vdots \\ \mathbf{A}(\mathbf{y}_m) \end{bmatrix}}_{\mathbf{A}(\mathbf{y})} \mathbf{x} + \underbrace{\begin{bmatrix} \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_m \end{bmatrix}}_{\mathbf{e}}$$

- $\mathbf{y}$ = registration, blurring, etc., parameters
- Goal: Improve parameters $\mathbf{y}$ and compute $\mathbf{x}$

29−th low resolution image



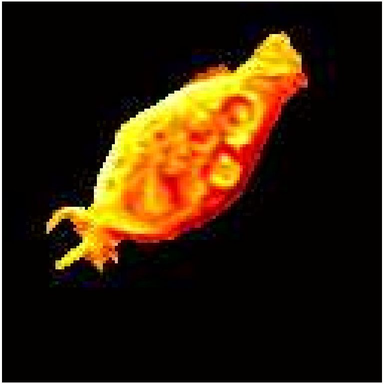
Application: Image Data Fusion

- $\mathbf{b}_j = \mathbf{A}(\mathbf{y}_j)\,\mathbf{x} + \mathbf{e}_j$
  (collected low resolution images)

$$\underbrace{\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_m \end{bmatrix}}_{\mathbf{b}} = \underbrace{\begin{bmatrix} \mathbf{A}(\mathbf{y}_1) \\ \vdots \\ \mathbf{A}(\mathbf{y}_m) \end{bmatrix}}_{\mathbf{A}(\mathbf{y})} \mathbf{x} + \underbrace{\begin{bmatrix} \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_m \end{bmatrix}}_{\mathbf{e}}$$

- $\mathbf{y}$ = registration, blurring, etc., parameters
- Goal: Improve parameters $\mathbf{y}$ and compute $\mathbf{x}$

Reconstructed high resolution image



Outline

The Linear Problem

Assume $\mathbf{A} = \mathbf{A}(\mathbf{y})$ is known exactly.

- We are given $\mathbf{A}$ and $\mathbf{b}$, where

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{e}$$

- $\mathbf{A}$ is an ill-conditioned matrix, and we do not know $\mathbf{e}$.
- We want to compute an approximation of $\mathbf{x}$.

- Bad idea:
  - $\mathbf{e}$ is small, so ignore it, and
  - use $\mathbf{x}_{\text{inv}} \approx \mathbf{A}^{-1}\mathbf{b}$

The Linear Problem

Assume $\mathbf{A} = \mathbf{A}(\mathbf{y})$ is known exactly.

- We are given $\mathbf{A}$ and $\mathbf{b}$, where

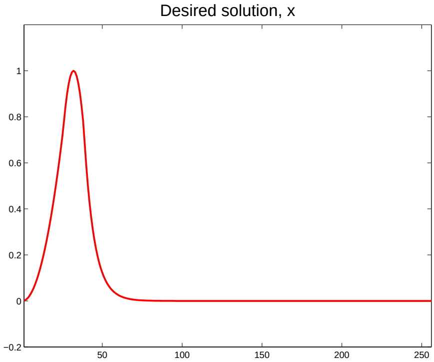
$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{e}$$

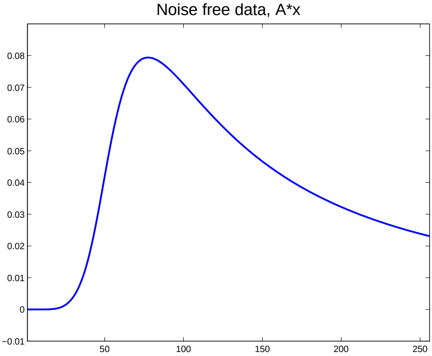
- $\mathbf{A}$ is an ill-conditioned matrix, and we do not know $\mathbf{e}$.
- We want to compute an approximation of $\mathbf{x}$.

- Bad idea:
  - $\mathbf{e}$ is small, so ignore it, and
  - use $\mathbf{x}_{\text{inv}} \approx \mathbf{A}^{-1}\mathbf{b}$

Example: Inverse Heat Equation

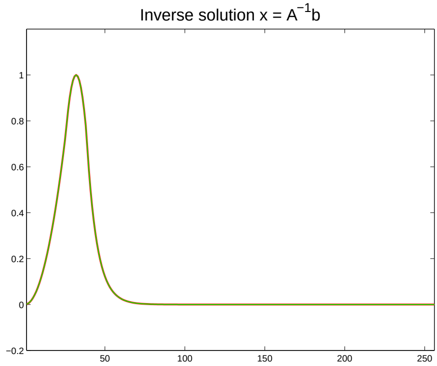
Regularization Tools test problem: heat.m
P. C. Hansen, www2.imm.dtu.dk/∼pch/Regutools

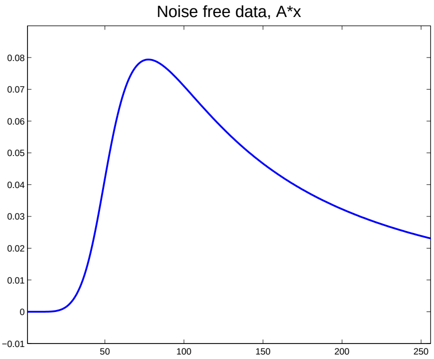




Example: Inverse Heat Equation

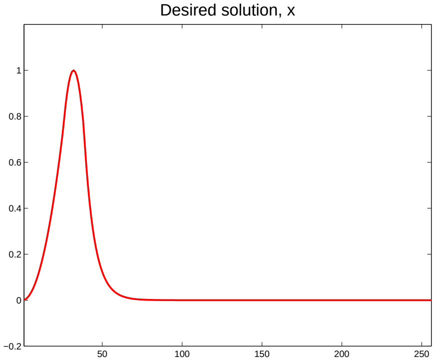
If $\mathbf{A}$ and $\mathbf{b}$ are known exactly,
can get an accurate reconstruction.

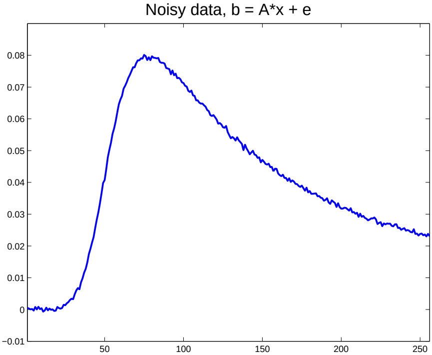




Example: Inverse Heat Equation

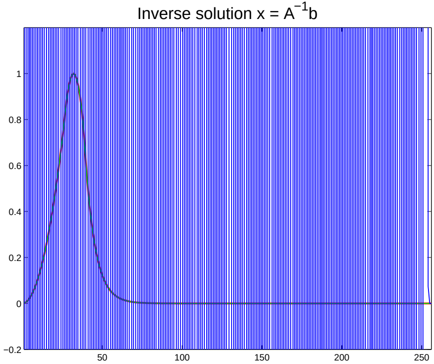
But, if $\mathbf{b}$ contains a small amount of noise,

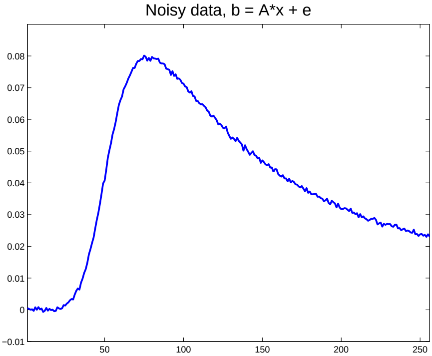




Example: Inverse Heat Equation

But, if $\mathbf{b}$ contains a small amount of noise, then we get a poor reconstruction!





SVD Analysis

An important linear algebra tool: Singular Value Decomposition

Let $\quad \mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \quad$ where

- $\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)\,, \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$
- $\mathbf{U}^T\mathbf{U} = \mathbf{I}\,, \quad \mathbf{V}^T\mathbf{V} = \mathbf{I}$
- $\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_n \end{bmatrix}$ (left singular vectors)
- $\mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix}$ (right singular vectors)

SVD Analysis

The naïve inverse solution can then be represented as:

$$\begin{aligned} \mathbf{x} &= \mathbf{A}^{-1}\mathbf{b} \\ &= \mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T\mathbf{b} \\ &= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T\mathbf{b}}{\sigma_i}\mathbf{v}_i \end{aligned}$$

SVD Analysis

The naïve inverse solution can then be represented as:

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{A}^{-1}(\mathbf{b} + \mathbf{e}) \\ &= \mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T(\mathbf{b} + \mathbf{e}) \\ &= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T(\mathbf{b} + \mathbf{e})}{\sigma_i}\mathbf{v}_i\end{aligned}$$

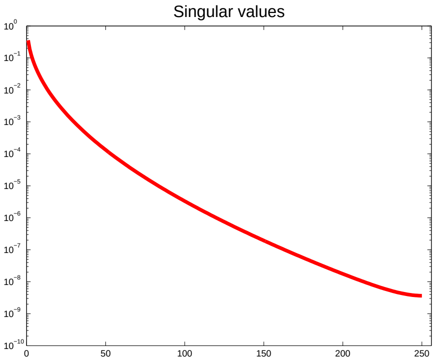
SVD Analysis

The naïve inverse solution can then be represented as:

$$
\begin{aligned}
\hat{\mathbf{x}} &= \mathbf{A}^{-1}(\mathbf{b} + \mathbf{e}) \\
&= \mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T(\mathbf{b} + \mathbf{e}) \\
&= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T(\mathbf{b} + \mathbf{e})}{\sigma_i}\mathbf{v}_i \\
&= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T\mathbf{b}}{\sigma_i}\mathbf{v}_i + \sum_{i=1}^{n} \frac{\mathbf{u}_i^T\mathbf{e}}{\sigma_i}\mathbf{v}_i \\
&= \mathbf{x} + \text{error}
\end{aligned}
$$

Example: Inverse Heat Equation

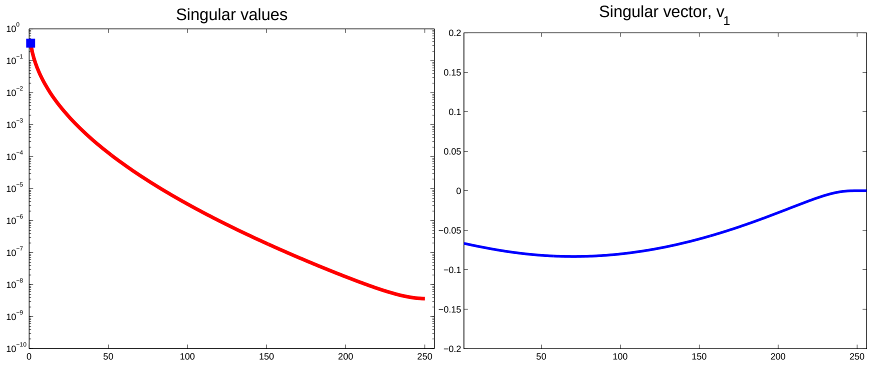
Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.



Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.

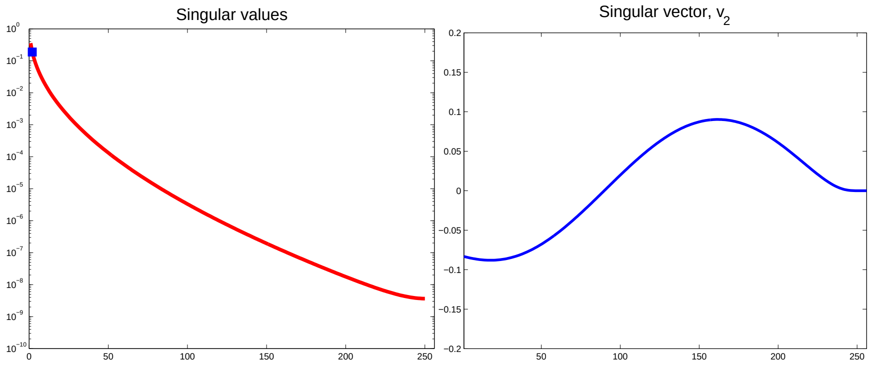
Large $\sigma_i \leftrightarrow$ smooth (low frequency) $\mathbf{v}_i$



Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.

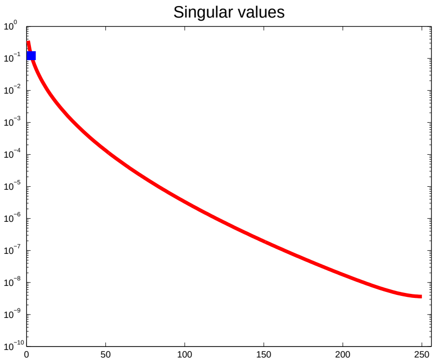
Large σ_i $\leftrightarrow$ smooth (low frequency) $\mathbf{v}_i$

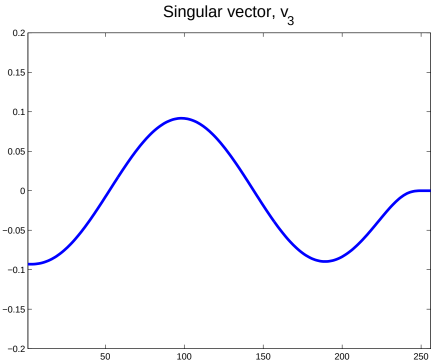


Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.

Large σ_i $\leftrightarrow$ smooth (low frequency) $\mathbf{v}_i$

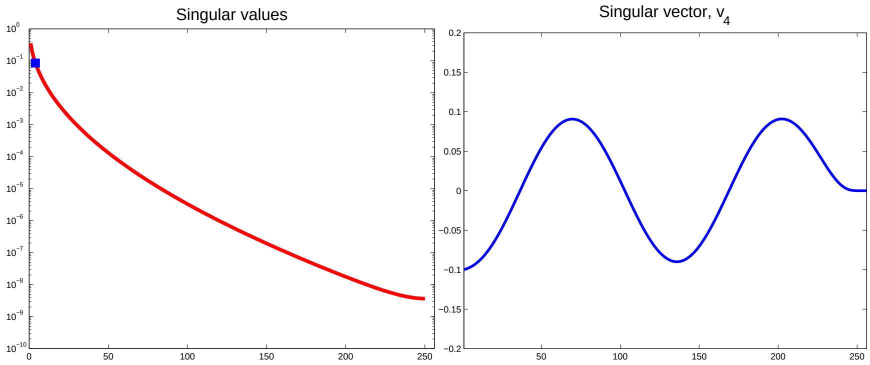




Example: Inverse Heat Equation

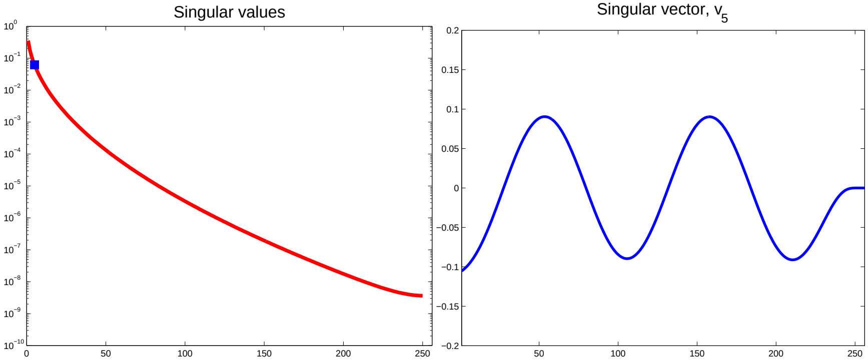
Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.

Large σ_i $\leftrightarrow$ smooth (low frequency) $\mathbf{v}_i$



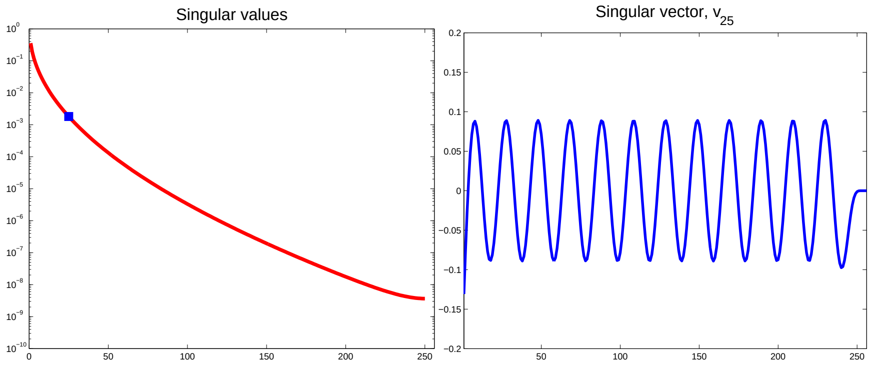
Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Large $\sigma_i \leftrightarrow$ smooth (low frequency) $\mathbf{v}_i$



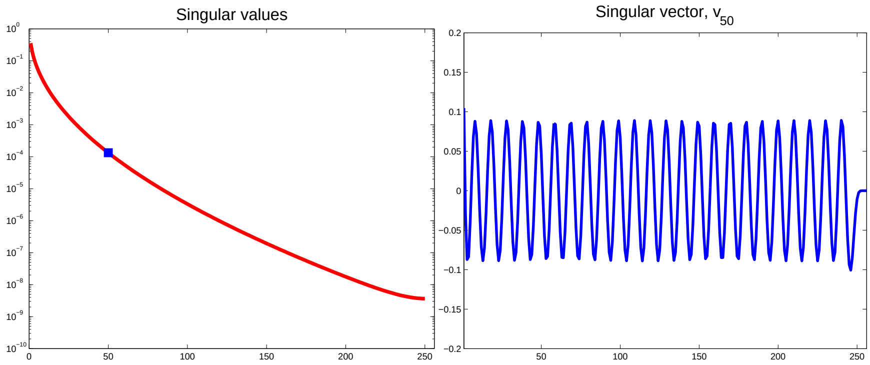
Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Small σ_i $\leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$



Example: Inverse Heat Equation

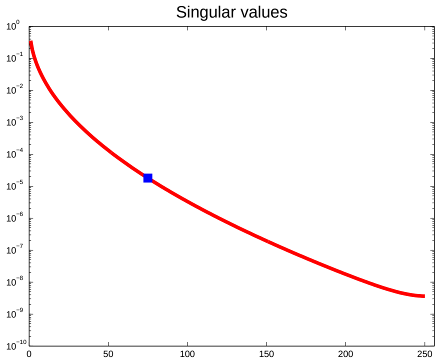
Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Small $\sigma_i \leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$

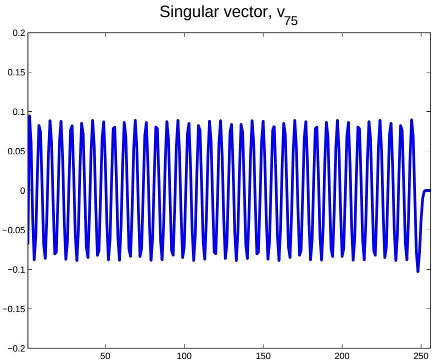


Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.

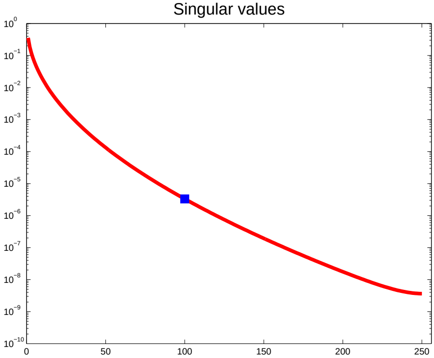
Small $\sigma_i \leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$

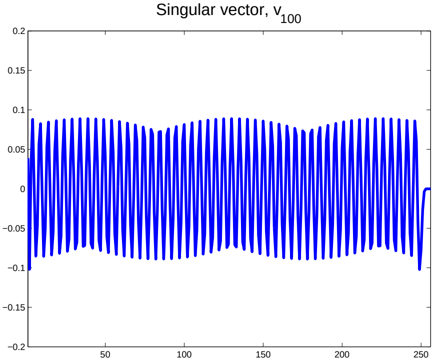




Example: Inverse Heat Equation

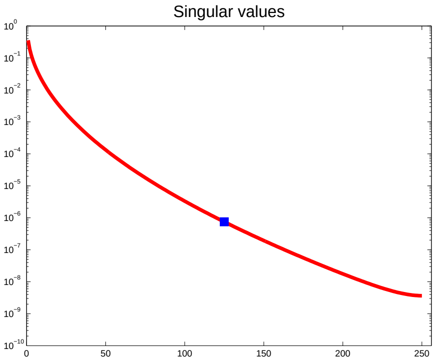
Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Small $\sigma_i \leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$

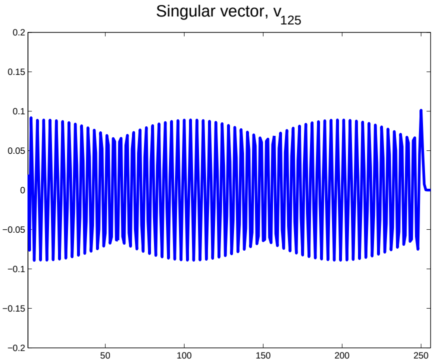




Example: Inverse Heat Equation

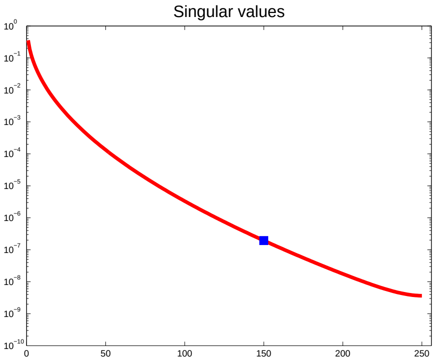
Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Small $\sigma_i \leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$

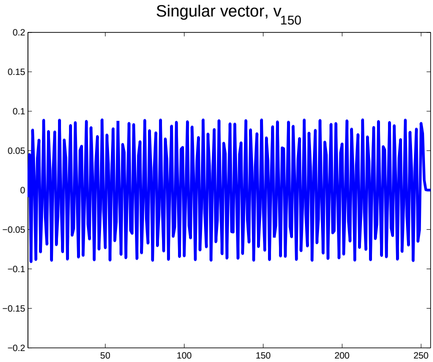




Example: Inverse Heat Equation

Error term depends on singular values σ_i and singular vectors $\mathbf{v}_i$.
Small $\sigma_i \leftrightarrow$ oscillating (high frequency) $\mathbf{v}_i$





SVD Analysis

The naïve inverse solution can then be represented as:

$$
\begin{aligned}
\hat{\mathbf{x}} &= \mathbf{A}^{-1}(\mathbf{b} + \mathbf{e}) \\
&= \mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T(\mathbf{b} + \mathbf{e}) \\
&= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T(\mathbf{b} + \mathbf{e})}{\sigma_i}\mathbf{v}_i \\
&= \sum_{i=1}^{n} \frac{\mathbf{u}_i^T\mathbf{b}}{\sigma_i}\mathbf{v}_i + \sum_{i=1}^{n} \frac{\mathbf{u}_i^T\mathbf{e}}{\sigma_i}\mathbf{v}_i \\
&= \mathbf{x} + \text{error}
\end{aligned}
$$

Regularization by Filtering

Basic Idea: Filter out effects of small singular values.
(Hansen, SIAM, 1997)

$$\mathbf{x}_{\text{reg}} = \mathbf{A}_{\text{reg}}^{-1}\mathbf{b} = \mathbf{V}\boldsymbol{\Phi}\boldsymbol{\Sigma}^{-1}\mathbf{U}^T\mathbf{b} = \sum_{i=1}^{n} \phi_i \frac{\mathbf{u}_i^T\mathbf{b}}{\sigma_i}\mathbf{v}_i\,,$$

where $\boldsymbol{\Phi} = \text{diag}(\phi_1, \phi_2, \ldots, \phi_n)$

The "filter factors" satisfy

$$\phi_i \approx \begin{cases} 1 & \text{if } \sigma_i \text{ is large} \\ 0 & \text{if } \sigma_i \text{ is small} \end{cases}$$

An Example: Tikhonov Regularization

$$\min_{\mathbf{x}} \left\{ \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda^2 \|\mathbf{x}\|_2^2 \right\} \quad \Leftrightarrow \quad \min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} \right\|_2^2$$

An Example: Tikhonov Regularization

$$\min_{\mathbf{x}} \left\{ \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda^2 \|\mathbf{x}\|_2^2 \right\} \quad \Leftrightarrow \quad \min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} \right\|_2^2$$

An equivalent SVD filtering formulation:

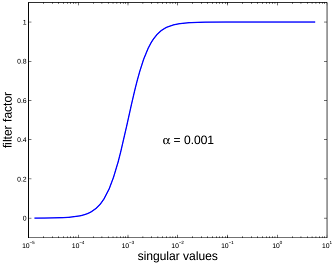
$$\mathbf{x}_{\text{tik}} = \sum_{i=1}^{n} \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i$$

An Example: Tikhonov Regularization

$$\min_{\mathbf{x}} \left\{ \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda^2 \|\mathbf{x}\|_2^2 \right\} \quad \Leftrightarrow \quad \min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} \right\|_2^2$$

An equivalent SVD filtering formulation:

$$\mathbf{x}_{\text{tik}} = \sum_{i=1}^{n} \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i$$



Choosing Regularization Parameters

Lots of choices: Generalized Cross Validation (GCV), L-curve, discrepancy principle, ...

Choosing Regularization Parameters

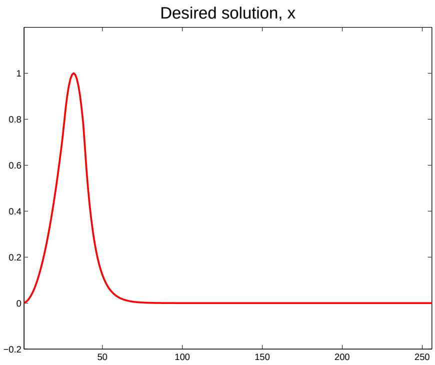
Lots of choices: Generalized Cross Validation (GCV), L-curve, discrepancy principle, ...

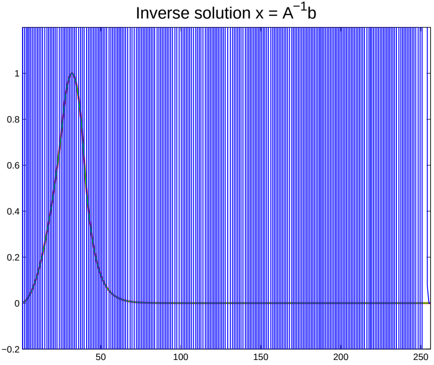
GCV and Tikhonov: Choose λ to minimize

$$\mathrm{GCV}(\lambda) = \frac{n \displaystyle\sum_{i=1}^{n} \left(\frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i^2 + \lambda^2} \right)^2}{\left(\displaystyle\sum_{i=1}^{n} \frac{1}{\sigma_i^2 + \lambda^2} \right)^2}$$

Example: Inverse Heat Equation

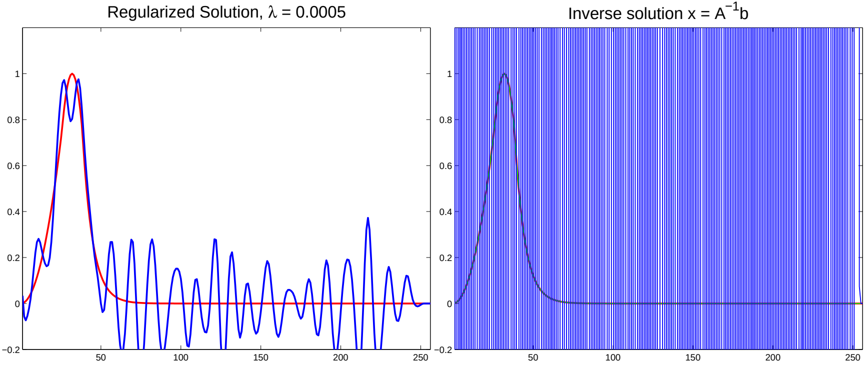
Reconstruction using Tikhonov reg. can be better than $\mathbf{x}_{\text{inv}}$.
Quality of reconstruction depends on λ.
But λ depends on $\mathbf{A}$ and $\mathbf{b}$.





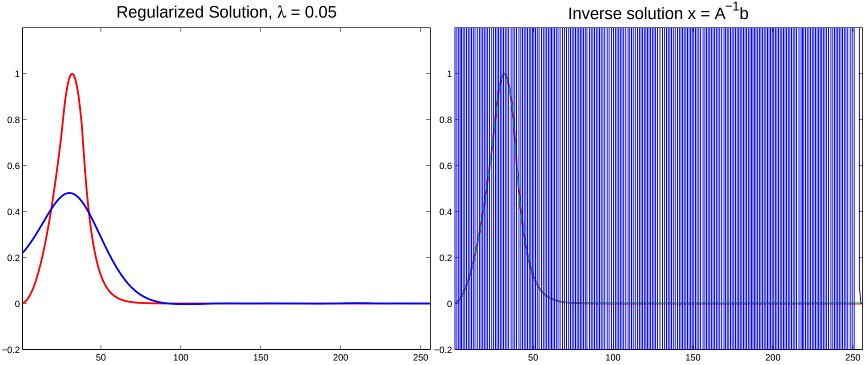
Example: Inverse Heat Equation

Reconstruction using Tikhonov reg. can be better than $\mathbf{x}_{\rm inv}$.
Quality of reconstruction depends on λ.
But λ depends on $\mathbf{A}$ and $\mathbf{b}$.



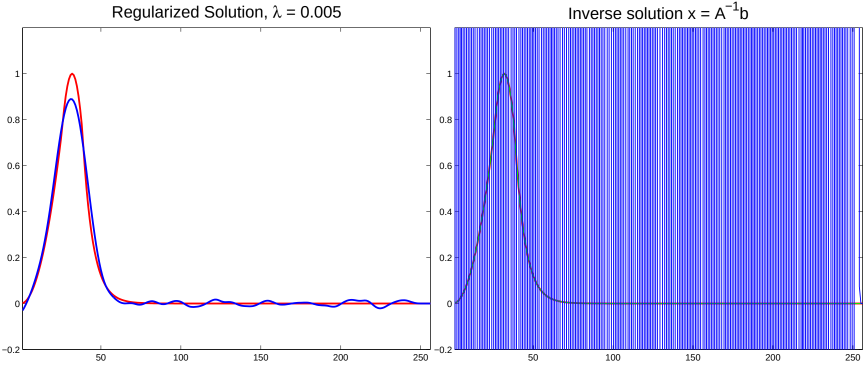
Example: Inverse Heat Equation

Reconstruction using Tikhonov reg. can be better than $\mathbf{x}_{\text{inv}}$.
Quality of reconstruction depends on λ.
But λ depends on $\mathbf{A}$ and $\mathbf{b}$.



Example: Inverse Heat Equation

Reconstruction using Tikhonov reg. can be better than $\mathbf{x}_{\mathrm{inv}}$.
Quality of reconstruction depends on λ.
But λ depends on $\mathbf{A}$ and $\mathbf{b}$.



Filtering for Large Scale Problems

Some remarks:

- For large matrices, computing SVD is expensive.
- SVD algorithms do not readily simplify for structured or sparse matrices.
- Alternative for large scale problems: LSQR iteration (Paige and Saunders, ACM TOMS, 1982)

Lanczos Bidiagonalization (LBD)

Given $\mathbf{A}$ and $\mathbf{b}$, for $k = 1, 2, \ldots$, compute

- $\mathbf{W}_k = \begin{bmatrix} \mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_k & \mathbf{w}_{k+1} \end{bmatrix}, \quad \mathbf{w}_1 = \mathbf{b}/||\mathbf{b}||$
- $\mathbf{Z}_k = \begin{bmatrix} \mathbf{z}_1 & \mathbf{z}_2 & \cdots & \mathbf{z}_k \end{bmatrix}$
- $\mathbf{B}_k = \begin{bmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_k & \alpha_k \\ & & & \beta_{k+1} \end{bmatrix}$

where $\mathbf{W}_k$ and $\mathbf{Z}_k$ have orthonormal columns, and

$$\begin{aligned} \mathbf{A}^T\mathbf{W}_k &= \mathbf{Z}_k\mathbf{B}_k^T + \alpha_{k+1}\mathbf{z}_{k+1}\mathbf{e}_{k+1}^T \\ \mathbf{A}\mathbf{Z}_k &= \mathbf{W}_k\mathbf{B}_k \end{aligned}$$

LBD and LSQR

At kth LBD iteration, use QR to solve *projected* LS problem:

$$\min_{\mathbf{x} \in R(\mathbf{Z}_k)} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 = \min_{\mathbf{f}} \|\mathbf{W}_k^T \mathbf{b} - \mathbf{B}_k \mathbf{f}\|_2^2 = \min_{\mathbf{f}} \|\beta \mathbf{e}_1 - \mathbf{B}_k \mathbf{f}\|_2^2$$

where $\mathbf{x}_k = \mathbf{Z}_k \mathbf{f}$

LBD and LSQR

At kth LBD iteration, use QR to solve *projected* LS problem:

$$\min_{\mathbf{x} \in R(\mathbf{Z}_k)} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 = \min_{\mathbf{f}} \|\mathbf{W}_k^T \mathbf{b} - \mathbf{B}_k \mathbf{f}\|_2^2 = \min_{\mathbf{f}} \|\beta \mathbf{e}_1 - \mathbf{B}_k \mathbf{f}\|_2^2$$

where $\mathbf{x}_k = \mathbf{Z}_k \mathbf{f}$

For our ill-posed inverse problems:

- Singular values of $\mathbf{B}_k$ converge to k largest sing. values of $\mathbf{A}$.
- Thus, $\mathbf{x}_k$ is in a subspace that approximates a subspace spanned by the large singular components of $\mathbf{A}$.
  - For $k < n$, $\mathbf{x}_k$ is a regularized solution.
  - $\mathbf{x}_n = \mathbf{x}_{\text{inv}} = \mathbf{A}^{-1}\mathbf{b}$ (bad approximation)

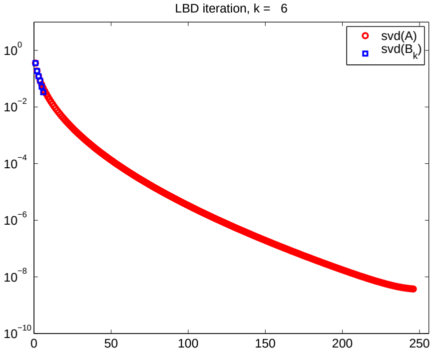
Example: Inverse Heat Equation

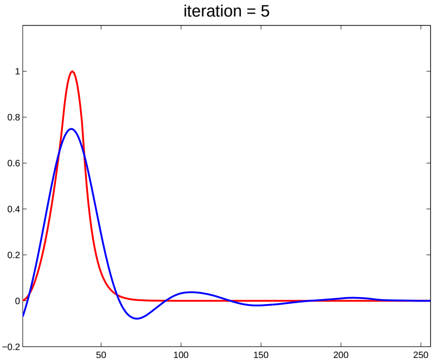
Singular values of $\mathbf{B}_k$ converge to large singular values of $\mathbf{A}$.

Thus, for early iterations k:

$$\begin{aligned} \mathbf{f} &= \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$

is a regularized reconstruction.



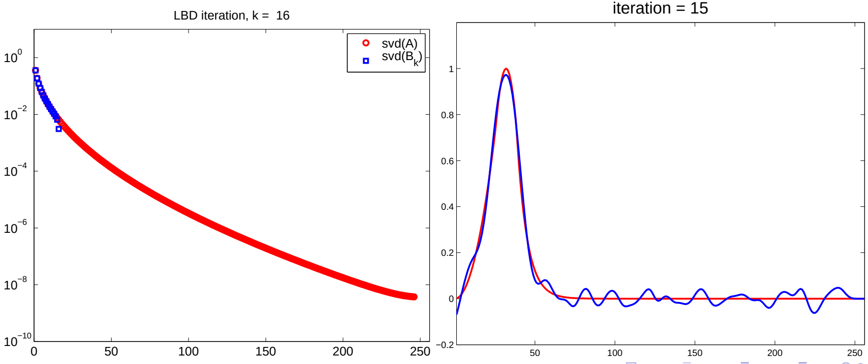


Example: Inverse Heat Equation

Singular values of $\mathbf{B}_k$ converge to large singular values of $\mathbf{A}$.

Thus, for early iterations k:
$$\begin{aligned} \mathbf{f} &= \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$

is a regularized reconstruction.



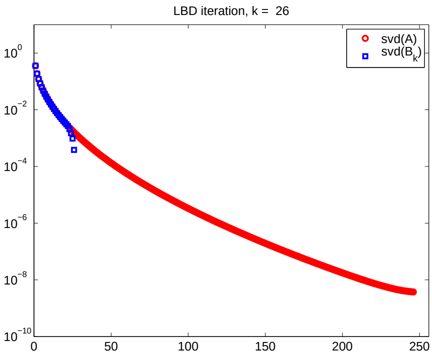
Example: Inverse Heat Equation

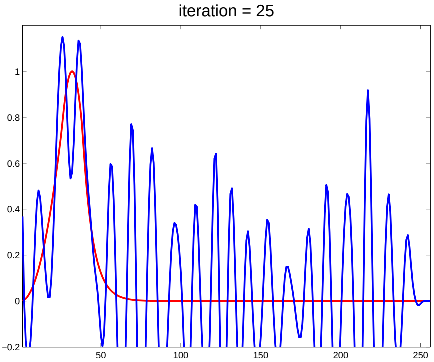
Singular values of $\mathbf{B}_k$ converge to large singular values of $\mathbf{A}$.

Thus, for later iterations k:

$$\begin{aligned} \mathbf{f} &= \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$

is a noisy reconstruction.





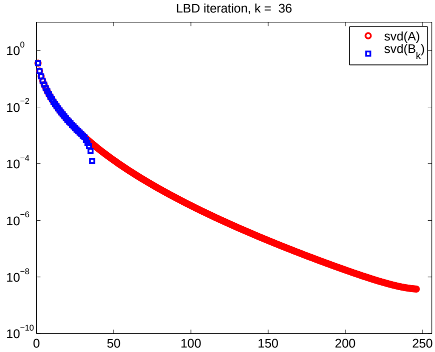
Example: Inverse Heat Equation

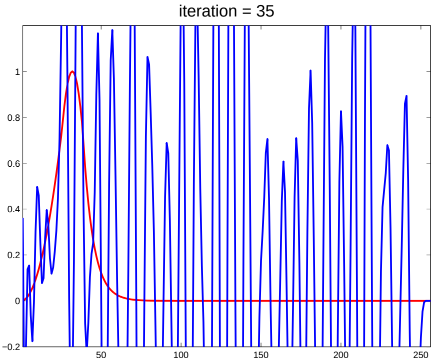
Singular values of $\mathbf{B}_k$ converge to large singular values of $\mathbf{A}$.

Thus, for later iterations k:

$$\begin{aligned} \mathbf{f} &= \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$

is a noisy reconstruction.





Lanczos Based Hybrid Methods

To avoid noisy reconstructions, embed regularization in LBD:

- O'Leary and Simmons, SISSC, 1981.
- Björck, BIT 1988.
- Björck, Grimme, and Van Dooren, BIT, 1994.
- Larsen, PhD Thesis, 1998.
- Hanke, BIT 2001.
- Kilmer and O'Leary, SIMAX, 2001.
- Kilmer, Hansen, Español, SISC 2007.
- Chung, N, O'Leary, ETNA 2007
  (HyBR Implementation)

Regularize the Projected Least Squares Problem

To stabilize convergence, regularize the projected problem:

$$\min_{\mathbf{f}} \left\| \begin{bmatrix} \beta\mathbf{e}_1 \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{B}_k \\ \lambda\mathbf{I} \end{bmatrix} \mathbf{f} \right\|_2^2$$

Note: $\mathbf{B}_k$ is very small compared to $\mathbf{A}$, so

- Can use "expensive" methods to choose λ (e.g., GCV)
- Very little regularization is needed in early iterations.
- GCV tends to choose too large λ for bidiagonal system. Our remedy: Use a weighted GCV (Chung, N, O'Leary, 2007)
- Can also use WGCV information to estimate stopping iteration (approach similar to Björck, Grimme, and Van Dooren, BIT, 1994).

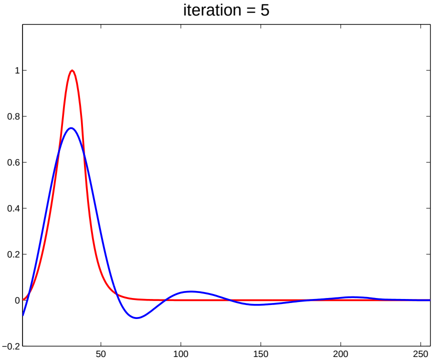
Example: Inverse Heat Equation

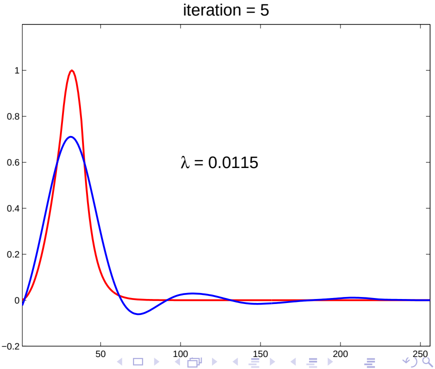
LSQR (no regularization)

$$\mathbf{f} = \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b}$$
$$\mathbf{x}_k = \mathbf{Z}_k \mathbf{f}$$

HyBR (Tikhonov regularization)

$$\mathbf{f} = \begin{bmatrix} \mathbf{B}_k \\ \lambda_k \mathbf{I} \end{bmatrix} \setminus \begin{bmatrix} \mathbf{W}_k \mathbf{b} \\ \mathbf{0} \end{bmatrix}$$
$$\mathbf{x}_k = \mathbf{Z}_k \mathbf{f}$$





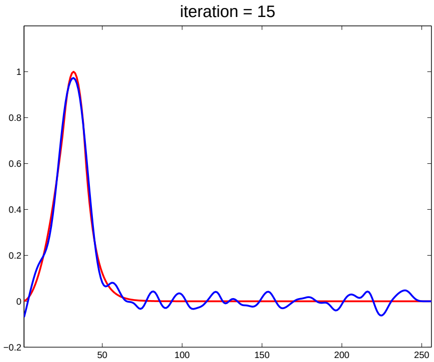
Example: Inverse Heat Equation

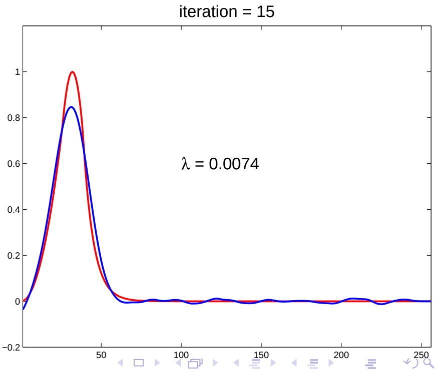
LSQR (no regularization)

$$\mathbf{f} = \mathbf{B}_k \setminus \mathbf{W}_k\mathbf{b}$$
$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{f}$$

HyBR (Tikhonov regularization)

$$\mathbf{f} = \begin{bmatrix} \mathbf{B}_k \\ \lambda_k\mathbf{I} \end{bmatrix} \setminus \begin{bmatrix} \mathbf{W}_k\mathbf{b} \\ \mathbf{0} \end{bmatrix}$$
$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{f}$$





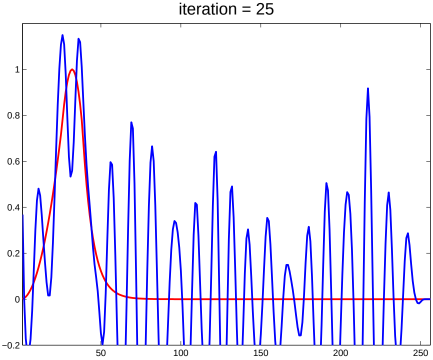
Example: Inverse Heat Equation

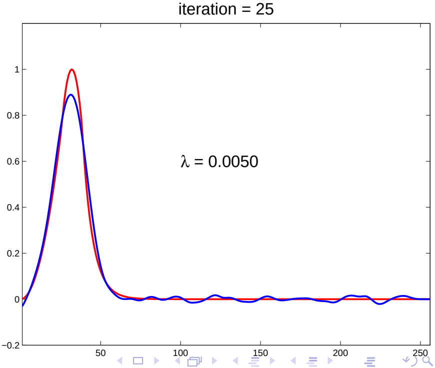
LSQR (no regularization)

$$\begin{aligned} \mathbf{f} &= \mathbf{B}_k \setminus \mathbf{W}_k \mathbf{b} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$

HyBR (Tikhonov regularization)

$$\begin{aligned} \mathbf{f} &= \begin{bmatrix} \mathbf{B}_k \\ \lambda_k \mathbf{I} \end{bmatrix} \Bigg\backslash \begin{bmatrix} \mathbf{W}_k \mathbf{b} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{x}_k &= \mathbf{Z}_k \mathbf{f} \end{aligned}$$





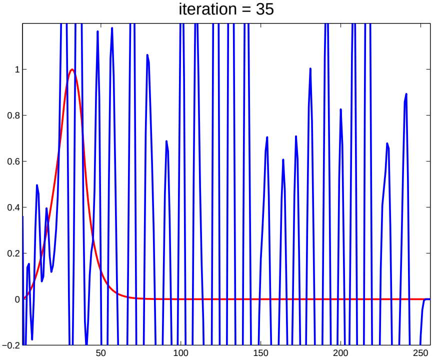
Example: Inverse Heat Equation

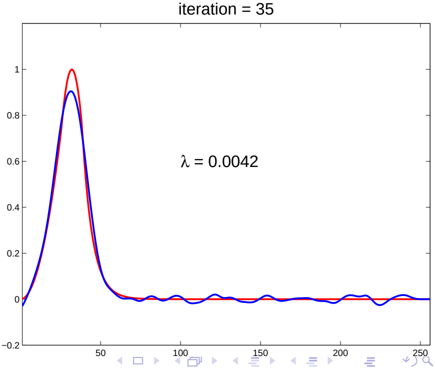
LSQR (no regularization)

$$\mathbf{f} = \mathbf{B}_k \setminus \mathbf{W}_k\mathbf{b}$$
$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{f}$$

HyBR (Tikhonov regularization)

$$\mathbf{f} = \begin{bmatrix} \mathbf{B}_k \\ \lambda_k \mathbf{I} \end{bmatrix} \Bigg\backslash \begin{bmatrix} \mathbf{W}_k\mathbf{b} \\ \mathbf{0} \end{bmatrix}$$
$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{f}$$





The Nonlinear Problem

- We want to find $\mathbf{x}$ and $\mathbf{y}$ so that

$$\mathbf{b} = \mathbf{A}(\mathbf{y})\mathbf{x} + \mathbf{e}$$

- With Tikhonov regularization, solve

$$\min_{\mathbf{x},\mathbf{y}} \left\| \begin{bmatrix} \mathbf{A}(\mathbf{y}) \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right\|_2^2$$

- As with linear problem, choosing a good regularization parameter λ is important.
- Problem is linear in $\mathbf{x}$, nonlinear in $\mathbf{y}$.
- $\mathbf{y} \in \mathcal{R}^p$, $\mathbf{x} \in \mathcal{R}^n$, with $p \ll n$.

Separable Nonlinear Least Squares

Variable Projection Method:

- Implicitly eliminate linear term.
- Optimize over nonlinear term.

Some general references:

Golub and Pereyra, SINUM 1973 (also IP 2003)
Kaufman, BIT 1975
Osborne, SINUM 1975 (also ETNA 2007)
Ruhe and Wedin, SIREV, 1980

How to apply to inverse problems?

Variable Projection Method

Instead of optimizing over both $\mathbf{x}$ and $\mathbf{y}$:

$$\min_{\mathbf{x},\mathbf{y}} \phi(\mathbf{x},\mathbf{y}) = \min_{\mathbf{x},\mathbf{y}} \left\| \begin{bmatrix} \mathbf{A}(\mathbf{y}) \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right\|_2^2$$

Let $\mathbf{x}(\mathbf{y})$ be solution of

$$\min_{\mathbf{x}} \phi(\mathbf{x},\mathbf{y}) = \min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{A}(\mathbf{y}) \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right\|_2^2$$

and then minimize the reduced cost functional:

$$\min_{\mathbf{y}} \psi(\mathbf{y})\,, \quad \psi(\mathbf{y}) = \phi(\mathbf{x}(\mathbf{y}),\mathbf{y})$$

Gauss-Newton Algorithm

choose initial $\mathbf{y}_0$

for $k = 0, 1, 2, \ldots$

$$\mathbf{x}_k = \arg\min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{A}(\mathbf{y}_k) \\ \lambda_k \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right\|_2$$

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}(\mathbf{y}_k)\,\mathbf{x}_k$$

$$\mathbf{d}_k = \arg\min_{\mathbf{d}} \|\mathbf{J}_\psi \mathbf{d} - \mathbf{r}_k\|_2$$

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{d}_k$$

end

Gauss-Newton Algorithm with HyBR

And we use HyBR to solve the linear subproblem:

choose initial $\mathbf{y}_0$

for $k = 0, 1, 2, \ldots$

$$\mathbf{x}_k = \text{HyBR}(\mathbf{A}(\mathbf{y}_k), \mathbf{b})$$

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}(\mathbf{y}_k)\,\mathbf{x}_k$$

$$\mathbf{d}_k = \arg\min_{\mathbf{d}} \|\mathbf{J}_\psi \mathbf{d} - \mathbf{r}_k\|_2$$

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{d}_k$$

end

Example: Image Deblurring

Matrix $\mathbf{A}(\mathbf{y})$ is defined by a PSF, which is in turn defined by parameters. Specifically:

$$\mathbf{A}(\mathbf{y}) = \mathbf{A}(\mathbf{P}(\mathbf{y}))$$

where

- $\mathbf{A}$ is 65536×65536, with entries given by $\mathbf{P}$.
- $\mathbf{P}$ is 256×256, with entries:

$$p_{ij} = \exp\left(\frac{(i-k)^2 s_2^2 - (j-l)^2 s_1^2 + 2(i-k)(j-l)\rho^2}{2 s_1^2 s_2^2 - 2\rho^4} \right)$$

- (k, l) is the PSF center (location of point source)
- $\mathbf{y}$ vector of unknown parameters:

$$\mathbf{y} = \begin{bmatrix} s_1 \\ s_2 \\ \rho \end{bmatrix}$$

Example: Image Deblurring

Can get analytical formula for Jacobian:

$$
\begin{aligned}
\mathbf{J}_\psi &= \frac{\partial}{\partial \mathbf{y}} \{ \mathbf{A}(\,\mathbf{P}(\mathbf{y})\,)\,\mathbf{x} \} \\
&= \frac{\partial}{\partial \mathbf{P}} \{ \mathbf{A}(\,\mathbf{P}(\mathbf{y})\,)\,\mathbf{x} \} \cdot \frac{\partial}{\partial \mathbf{y}} \{ \mathbf{P}(\mathbf{y}) \} \\
&= \mathbf{A}(\mathbf{X}) \cdot \frac{\partial}{\partial \mathbf{y}} \{ \mathbf{P}(\mathbf{y}) \}
\end{aligned}
$$

where $\mathbf{x} = \mathrm{vec}(\mathbf{X})$.

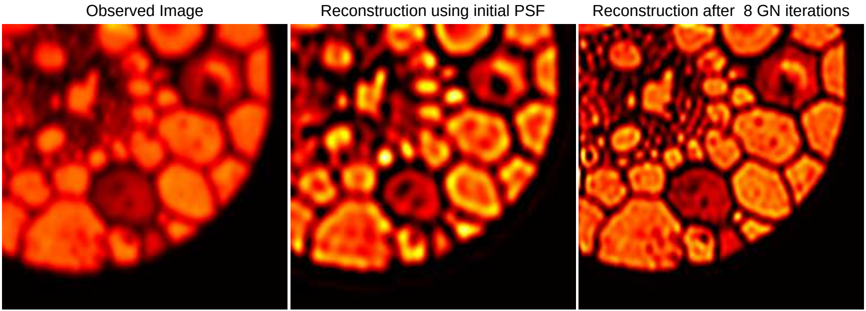
Though in this example, finite difference approximation of $\mathbf{J}_\psi$ works very well.

Example: Image Deblurring

Gauss-Newton Iteration History

G-N Iteration	$\Delta\mathbf{y}$	λ
0	0.5716	0.1685
1	0.3345	0.1223
2	0.2192	0.0985
3	0.1473	0.0804
4	0.1006	0.0715
5	0.0648	0.0676
6	0.0355	0.0657
7	0.0144	0.0650

Example: Image Deblurring



Concluding Remarks

- Imaging applications require solving challenging inverse problems.
- Separable nonlinear least squares models exploit high level structure.
- Hybrid methods are efficient solvers for large scale linear inverse problems.
  - Automatic estimation of regularization parameter.
  - Automatic estimation of stopping iteration.
- Hybrid methods can be effective linear solvers for nonlinear problems.

Questions?

- Other methods to choose regularization parameters?
- Other regularization methods (e.g., total variation)?
- Sparse (in some basis) reconstructions?
- MATLAB Codes and Data?

  www.mathcs.emory.edu/∼nagy/WGCV
  www.mathcs.emory.edu/∼nagy/RestoreTools
  www2.imm.dtu.dk/∼pch/HNO
  www2.imm.dtu.dk/∼pch/Regutools