Inverse problems and application to image deblurring

Oleg Makhnin
New Mexico Tech
Dept. of Mathematics

October 29, 2010

Intro: inverse problems

Tikhonov Regularization

Image deblurring/ deconvolution

CGLS solution, use of ℓ_1 norm

Fourier approach

Beyond

Intro: inverse problems

Tikhonov Regularization

Image deblurring/ deconvolution

CGLS solution, use of ℓ_1 norm

Fourier approach

Beyond

Intro: inverse problems

Tikhonov Regularization

Image deblurring/ deconvolution

CGLS solution, use of ℓ_1 norm

Fourier approach

Beyond

Intro: inverse problems

Tikhonov Regularization

Image deblurring/ deconvolution

CGLS solution, use of ℓ_1 norm

Fourier approach

Beyond

Intro: inverse problems

Forward problem

$$G(\mathbf{x}) = \mathbf{d}$$

where $\mathbf{x}$ is the "state" vector (typically hidden) and $\mathbf{d}$ is your data vector.
G is assumed known. Its estimation is an additional challenge.
The equation may be overdefined (in this case the solution is fitted in the least squares sense), or

underdefined in which case we have to pick "the best" solution, and some features of "true" $\mathbf{x}$ might be lost.

Intro: inverse problems

Forward problem

$$G(\mathbf{x}) = \mathbf{d}$$

where $\mathbf{x}$ is the "state" vector (typically hidden) and $\mathbf{d}$ is your data vector.

G is assumed known. Its estimation is an additional challenge.

The equation may be overdefined (in this case the solution is fitted in the least squares sense), or

underdefined in which case we have to pick "the best" solution, and some features of "true" $\mathbf{x}$ might be lost.

Linear problem: when $G(\cdot)$ is given by the multipication by the matrix G.

Intro: inverse problems

Forward problem

$$G(\mathbf{x}) = \mathbf{d}$$

where $\mathbf{x}$ is the "state" vector (typically hidden) and $\mathbf{d}$ is your data vector.

G is assumed known. Its estimation is an additional challenge.

The equation may be overdefined (in this case the solution is fitted in the least squares sense), or

underdefined in which case we have to pick "the best" solution, and some features of "true" $\mathbf{x}$ might be lost.

Linear problem: when $G(\cdot)$ is given by the multipication by the matrix G.

Parabola example.

Intro: inverse problems

Inverse problem:

$$\mathbf{x} = G^{-1}\mathbf{d} \qquad ??$$

What if G is not square, or square but not invertible?

Intro: inverse problems

Inverse problem:

$$\mathbf{x} = G^{-1}\mathbf{d} \qquad ??$$

What if G is not square, or square but not invertible?

In overdefined case we can solve approximately, which leads to well-known

$$\hat{\mathbf{x}} = (G^T G)^{-1} G^T \mathbf{d}$$

Intro: inverse problems

Inverse problem:

$$\mathbf{x} = G^{-1}\mathbf{d} \qquad ??$$

What if G is not square, or square but not invertible?

In overdefined case we can solve approximately, which leads to well-known

$$\hat{\mathbf{x}} = (G^T G)^{-1} G^T \mathbf{d}$$

In general, let's say

$$\mathbf{x} = G^{\sharp}\mathbf{d}$$

where $G^{\sharp}$ is *pseudoinverse* of G.

Intro: inverse problems

Inverse problem:

$$\mathbf{x} = G^{-1}\mathbf{d} \qquad ??$$

What if G is not square, or square but not invertible?

In overdefined case we can solve approximately, which leads to well-known

$$\hat{\mathbf{x}} = (G^T G)^{-1} G^T \mathbf{d}$$

In general, let's say

$$\mathbf{x} = G^{\sharp}\mathbf{d}$$

where $G^{\sharp}$ is *pseudoinverse* of G.

However, noise is usually present: $G\mathbf{x} = \mathbf{d} + \mathbf{e}$
and this complicates matters

Intro: ill-posed problems

Ill-posedness: Two Ellipses pic

Intro: ill-posed problems

Ill-posedness: Two Ellipses pic

Thus, the need to regularize the solution, i.e. pick $G^\sharp$ which is not quite the inverse ($G^\sharp G \neq I$). This is known as "noise suppression" in signal processing.

Intro: ill-posed problems

Ill-posedness: Two Ellipses pic

Thus, the need to regularize the solution, i.e. pick $G^\sharp$ which is not quite the inverse ($G^\sharp G \neq I$). This is known as "noise suppression" in signal processing.

Tikhonov regularization: instead of optimization problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|G\mathbf{x} - \mathbf{d}\|^2$$

solve the problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left[\ \|G\mathbf{x} - \mathbf{d}\|^2 + \alpha^2 \|L\mathbf{x}\|^2\ \right]$$

where the latter is a penalty term ($\alpha =$ some scalar parameter).

Tikhonov reg.: $\min\left[\|G\mathbf{x}-\mathbf{d}\|^2+\alpha^2\|L\mathbf{x}\|^2\right]$

"0-order Tikhonov regularization" when $L = I$
i.e. punish "large" solutions.

Tikhonov reg.: $\min \left[\|G\mathbf{x} - \mathbf{d}\|^2 + \alpha^2 \|L\mathbf{x}\|^2 \right]$

"0-order Tikhonov regularization" when $L = I$

i.e. punish "large" solutions.

Equivalent to the extended Least Squares problem

$$\begin{bmatrix} G \\ \alpha L \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}$$

Tikhonov reg.: $\min \left[\|G\mathbf{x} - \mathbf{d}\|^2 + \alpha^2 \|L\mathbf{x}\|^2 \right]$

"0-order Tikhonov regularization" when $L = I$

i.e. punish "large" solutions.

Equivalent to the extended Least Squares problem

$$\begin{bmatrix} G \\ \alpha L \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}$$

Tradeoff between "fit" and "solution norm". Pic: L-curve

Tikhonov reg.: $\min \left[\|G\mathbf{x} - \mathbf{d}\|^2 + \alpha^2 \|L\mathbf{x}\|^2 \right]$

"0-order Tikhonov regularization" when $L = I$

i.e. punish "large" solutions.

Equivalent to the extended Least Squares problem

$$\begin{bmatrix} G \\ \alpha L \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}$$

Tradeoff between "fit" and "solution norm". Pic: L-curve

Demo: Filtering a 1-D signal

Image deblurring

aka Deconvolution

Forward problem: blurring the image **x** by applying some matrix G.
What G does: transforms a single pixel into a spread-out shape described by PSF (point-spread function).

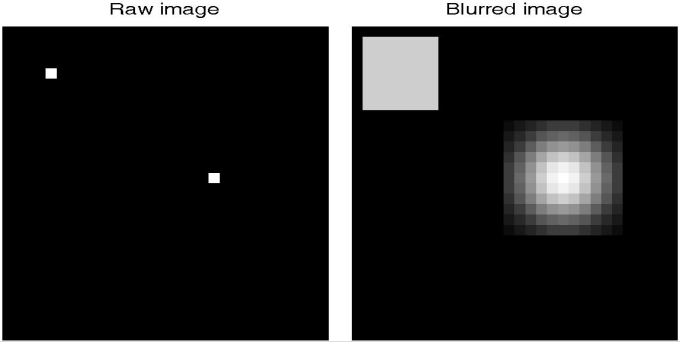


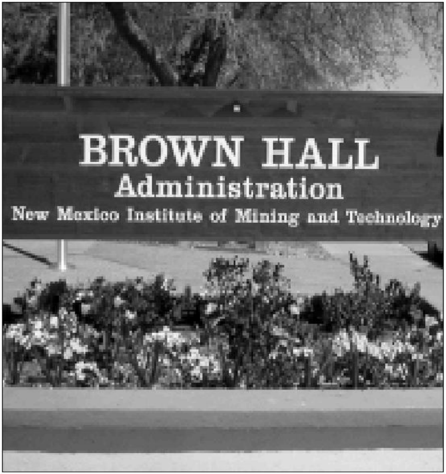
Image deblurring

The image on the rectangular grid is reshaped into the vector $\mathbf{x}$. Assume that PSF is known.

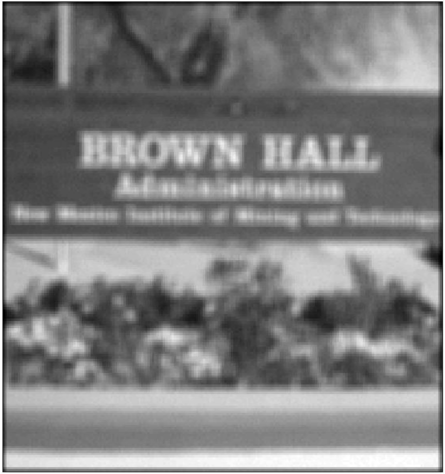
Apply CGLS (Conjugate Gradients for Least Squares) - iterative method with no need to explicitly form G.

In fact, stopping CGLS early acts as a regularizer of sorts.

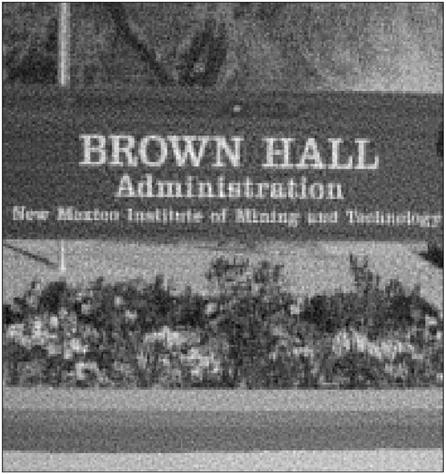
Raw image



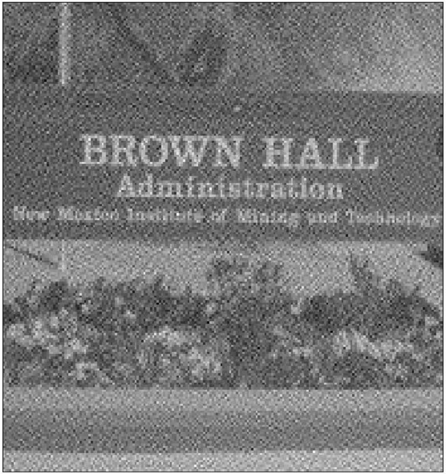
Blurred image with noise



CGLS solution after 30 iter



CGLS solution after 100 iter



Looks promising: use $L =$ finite difference matrix. This supresses wiggly behavior.

Another idea: to reduce smoothing (allow for sharper transitions), we might use ℓ_1 norm instead of ℓ_2 norm. To find the solution, try IRLS = iteratively reweighted least squares. It pays extra attention to rows of G that produce large residuals.

Again, revisit the toy 1d problem

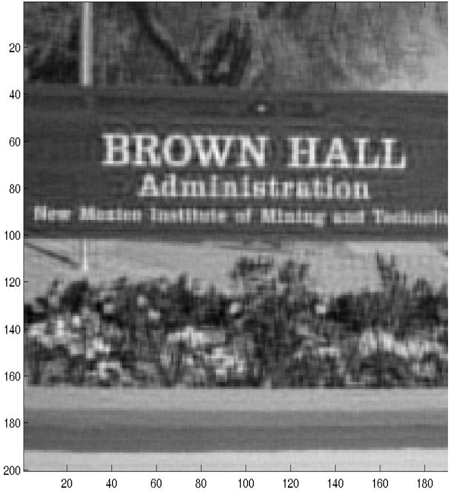
Looks promising: use $L =$ finite difference matrix. This supresses wiggly behavior.

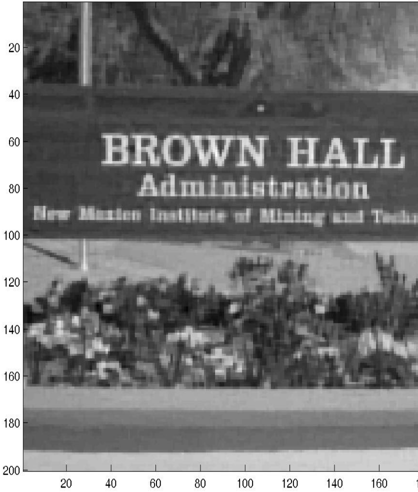
Another idea: to reduce smoothing (allow for sharper transitions), we might use ℓ_1 norm instead of ℓ_2 norm. To find the solution, try IRLS = iteratively reweighted least squares. It pays extra attention to rows of G that produce large residuals.

Again, revisit the toy 1d problem

So let's use IRLS together with CGLS to iterate towards a better estimate (one iteration inside the other)

Iterations 1 and 3:





Fourier methods

FFT (fast Fourier transform) can be used to do blurring, really fast.

Multiplication in frequency domain = convolution in **x**-domain.

Inverse Fourier transform does something akin to Tikhonov regularization. (After we suppress higher harmonics.)

But: how do we employ the penalty term/ ℓ_1 norm for it? (see the Gibbs phenomenon)

Beyond

What if the PSF is unknown? "Blind deconvolution"

Some recent progress has been made:

- Shan, Jia and Agarwala, 2008. *High-quality Motion Deblurring from a Single Image.* ACM Trans. Graph. 27, 3
- Levin, Weiss, Durand and W. T. Freeman. *Understanding and evaluating blind deconvolution algorithms.* IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), June 2009.

Both assume that PSF is the same at each location

Beyond

Other issues:

- Denoising
- Image inpainting

Beyond

Other issues:

- Denoising
- Image inpainting

Possible research projects:

- Can we use Fourier methods? If yes, how do we regularize them?
- Allowing for PSF to vary between locations

QUESTIONS?

THANK YOU!

see

www.nmt.edu/~olegm/talks/Deblur2