Linear Algebra and Deconvolution

M D Sacchi

SAIG Spring & Summer Lectures 2020
Physics, University of Alberta

Outline

- ☺ Notation
- ☺ Matrix Multiplication
- ☺ Norms
- ☺ Least-squares problems
  - Overdetermined Problems
  - Underdetermined Problems
- ☺ Regularization
- ☺ Convolution
- ☺ Deconvolution and Regularized Deconvolution

Notation

We define matrices by uppercase bold fonts and vectors by lowercase bold fonts. Upper or lowercase non-bold fonts denote scalars.

- α — Scalar
- a — Scalar
- $\mathbf{x}$ — Vector
- $\mathbf{A}$ — Matrix
- x_i — Element of a vector (a scalar)
- M_{ij} — Element of a matrix (a scalar)
- $f(\mathbf{x})$ — Scalar function with vector argument
- $u = g(\beta)$ — Scalar function with scalar argument

Notation

Pay attention to notation and consistency of mathematical expressions

- $c_i = \sum_i a_i b_i$ ☹
- $c = \sum_i a_i b_i$ ☺
- $c_j = \sum_i a_i b_{i,j}$ ☺
- $c_i = \sum_i a_i b_{i,j}$ ☹
- $c_{i,j} = \sum_i a_i b_{i,j}$ ☹
- $c = \|\mathbf{x}\|_2^2$ ☺
- $c = \|x_i\|_2^2$ ☹
- $\mathbf{x}_i$ ☹
- $c = \|\mathbf{x}_i\|_2^2$ ☹
- $M_{i,j} = \mathbf{x}$ ☹

Matrix Multiplication

In matrix-vector form:

$$\mathbf{d} = \mathbf{Am}$$

$\mathbf{d}$: $N \times 1$ vector
$\mathbf{A}$: $N \times M$ vector
$\mathbf{m}$: $M \times 1$ vector

In index form:

$$d_i = \sum_{j=1}^{M} A_{i,j}\, m_j \qquad i = 1 \ldots N$$

Matrix Multiplication

Always check size of vectors and matrices. Examples:

$\mathbf{u}$ and $\mathbf{m}$ are $M \times 1$, $\mathbf{A}$ is $N \times M$

$$\underbrace{\mathbf{d}}_{N\times 1} = \underbrace{\mathbf{A}}_{N\times M} \underbrace{\mathbf{m}}_{M\times 1}$$ A vector

$$\underbrace{\alpha}_{1\times 1} = \underbrace{\mathbf{u}^T}_{1\times M} \underbrace{\mathbf{m}}_{M\times 1}$$ A scalar

$$\underbrace{\mathbf{v}}_{M\times 1} = \underbrace{\mathbf{A}^T}_{M\times N} \underbrace{\mathbf{d}}_{N\times 1}$$ A vector

Matrix Multiplication

Check size of vectors and matrices:

$$\underbrace{\mathbf{y}}_{N\times 1} = \underbrace{\mathbf{A}}_{N\times M}\underbrace{\mathbf{x}}_{M\times 1}$$

$$\underbrace{\mathbf{A}^T}_{M\times N}\underbrace{\mathbf{y}}_{N\times 1} = \underbrace{\mathbf{A}^T}_{M\times N}\underbrace{\mathbf{A}}_{N\times M}\underbrace{\mathbf{x}}_{M\times 1}$$

if we let $\mathbf{g} = \mathbf{A}^T\mathbf{y}$ and $\mathbf{B} = \mathbf{A}^T\mathbf{A}$ then,

$$\underbrace{\mathbf{g}}_{M\times 1} = \underbrace{\mathbf{B}}_{M\times M}\underbrace{\mathbf{x}}_{M\times 1}$$

Norms

l_2 norm of a vector to avoid programming problems:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{M} x_i^2}$$

We usually work with the l_2^2:

$$\|\mathbf{x}\|_2^2 = \sum_{i=1}^{M} x_i^2$$

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^T\mathbf{x} = \begin{pmatrix} x_1 \; x_2 \; x_3 \ldots x_M \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_M \end{pmatrix}$$

Clearly, l_2 is a scalar.

Norms

Example:

$$\mathbf{x} = \begin{pmatrix} 2 \\ 3 \\ -2 \end{pmatrix}$$

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^T \mathbf{x} = \begin{pmatrix} 2 & 3 & -2 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ -2 \end{pmatrix} = 2^2 + 3^2 + (-2)^2 = 17$$

$$\|\mathbf{x}\|_2^2 = \sum_{i=1}^{3} x_i^2 = 2^2 + 3^2 + (-2)^2 = 17$$

Least-squares problem

Also consider a system where observations are related to model parameters via the following expression

$$\mathbf{d} \approx \mathbf{Am}$$

where $N > M$ (Overdetermined problems).
The latter can be written as follow

$$\mathbf{d} = \mathbf{Am} + \mathbf{e}$$

Where if $\mathbf{d}_0 = \mathbf{Am}$ is the ideal data, then

$$\mathbf{d} = \mathbf{d}_0 + \mathbf{e}$$

Task of the method of least-squares is to find the solution **m** that "best honours" the data **d**.

Least-squares problem

"Best" is a subjective word and we need an objective criterion. In the least-squares method we minimize the sum of the residuals

$$J = \|\mathbf{e}\|^2 = \sum_{i=1}^{N} e_i^2$$

J is call the cost function or objective function, a scalar function

$$J = \sum_i f(e_i)$$

where $f(\cdot) = (\cdot)^2$. Remember that you could change f to measure the error e_i in a different way.

Least-squares problem

Get use to use describe cost functions with different notation

$$J = \|\mathbf{e}\|^2 = \mathbf{e}^T\mathbf{e} = \sum_{i=1}^{N} e_i^2$$

Recall that $\mathbf{d} = \mathbf{A}\mathbf{m} + \mathbf{e}$. Then

$$J = \|\mathbf{d} - \mathbf{A}\mathbf{m}\|_2^2$$

which shows that J is a function of the unknown $\mathbf{m}$.

Least-squares problem

Notation again. We often say that we minimize J respect to $\mathbf{m}$ which is equivalent to finding the solution of the following problem

$$\frac{\partial J}{\partial \mathbf{m}} = \mathbf{0}^1$$

The solution of the latter is

$$\mathbf{A}^T\mathbf{A}\mathbf{m} = \mathbf{A}^T\mathbf{d}$$

From where you can do the following

$$\widehat{\mathbf{m}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{d}$$

$$\widehat{\mathbf{d}} = \mathbf{A}\widehat{\mathbf{m}} \qquad \text{and} \quad \widehat{\mathbf{e}} = \mathbf{d} - \widehat{\mathbf{d}}$$

$\widehat{\cdot}$ indicates estimated solution, estimated data or estimated error.

---

1 Notice I made **0** Bold!

Example

Let's assume we want to fit one period T years period to a time series. We use the following model

$$s(t_k) \approx c_0 + c\,sin(2\pi t_k/T + \phi) \qquad t_k = (k-1)\Delta t \qquad k = 1, \ldots N$$

We eliminate the phase[2]

$$s(t_k) \approx c_0 + A\,sin(2\pi t_k/T) + B\,cos(2\pi t_k/T)$$

The unknowns are c_0, A and B.

$$\underbrace{\begin{pmatrix} s(t_1) \\ s(t_2) \\ s(t_3) \\ \vdots \\ s(t_N) \end{pmatrix}}_{\mathbf{d}} \approx \underbrace{\begin{pmatrix} 1 & sin(2\pi t_1/T) & cos(2\pi t_1/T) \\ 1 & sin(2\pi t_2/T) & cos(2\pi t_2/T) \\ 1 & sin(2\pi t_3/T) & cos(2\pi t_3/T) \\ \vdots & \vdots & \vdots \\ 1 & sin(2\pi t_N/T) & cos(2\pi t_N/T) \end{pmatrix}}_{\mathbf{G}} \underbrace{\begin{pmatrix} c_0 \\ A \\ B \end{pmatrix}}_{\mathbf{m}}$$

$$\widehat{\mathbf{m}} = (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{d} \quad \widehat{c}_0 = \widehat{m}(1), \widehat{A} = \widehat{m}(2), \widehat{B} = \widehat{m}(3)$$

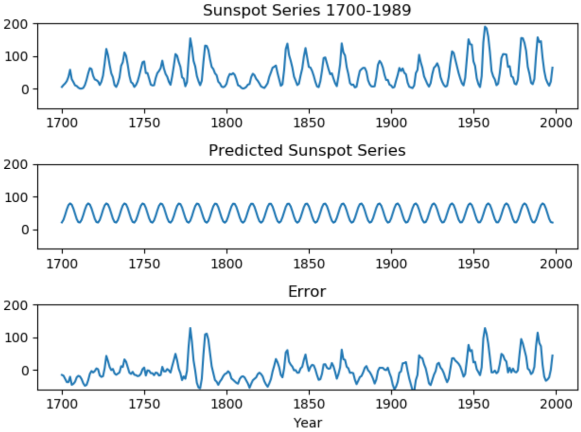
[2] $sin(x+y) = sin(x)cos(y) + cos(x)sin(y)$

Example

- Read data
- Form Matrix **G**
- Compute LS solution $\widehat{c}_0 = \widehat{m}(1), \widehat{A} = \widehat{m}(2), \widehat{B} = \widehat{m}(3)$
- Predict data $\widehat{s}(t_k) = \widehat{c}_0 + \widehat{A} sin(2\pi t_k / T) + \widehat{B} cos(2\pi t_k / T)$
- Compute Error $e(t_k) = s(t_k) - \widehat{s}(t_k)$

Example

$T = 11.04$ years



Not a nice result!

Example

Solution with more periods

$$s(t_k) \approx c_0 + \sum_{n=1}^{P} c_n \, sin(2\pi t_k / T_n + \phi_n) \qquad t_k = (k-1)\,\Delta t \qquad k = 1, \ldots N$$

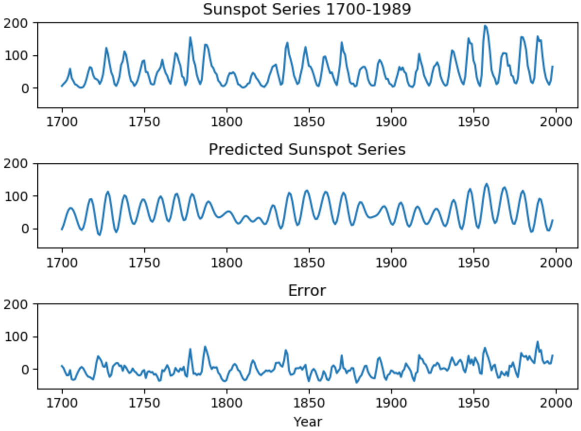
Again, we eliminate the phase

$$s(t_k) \approx c_0 + \sum_{n=1}^{P} A_n \, sin(2\pi t_k / T_n) + \sum_{n=1}^{P} B_n \, cos(2\pi t_k / T_n)$$

The unknowns are c_0, A_n and B_n. Total number of unknowns $N = 2P + 1$.

Example

We now fit 7 periods
$T = [11.04, 9.97, 98.33, 10.53, 11.92, 8.48, 59.81]$ years



Example

$f\,[c/y]$	$T\,[y]$	$\widehat{A}$	$\widehat{B}$
0.090579	11.04	7.0724	-26.235
0.100301	9.97	7.0131	-18.569
0.010169	98.33	-5.8638	15.983
0.094966	10.53	-6.7136	12.929
0.083892	11.92	6.3522	12.1131
0.117925	8.48	-10.6224	-2.2551
0.016719	59.81	-5.7533	5.4210
$\widehat{c}_0 =$	49.771		

Convolution

Convolution between two series

$$s_t = (w * r)_t = \sum_k w_{t-k} r_k$$

can be written in Matrix-times-vector form

$$\begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \end{pmatrix} = \begin{pmatrix} w_1 & 0 & 0 & 0 \\ w_2 & w_1 & 0 & 0 \\ w_3 & w_2 & w_1 & 0 \\ 0 & w_3 & w_2 & w_1 \\ 0 & 0 & w_3 & w_2 \\ 0 & 0 & 0 & w_3 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix}$$

$$\mathbf{s} = \mathbf{W}\,\mathbf{r}$$

Wavelet **w** is length N_w, Reflectivity **r** is length N_r then the seismogram **s** is length $N_s = N_w + N_r - 1$. We call **W** the convolution matrix. The problem is overdetermined because the matrix is of size $N \times M$ with $N > M$

Deconvolution

- This is time deconvolution (Frequency domain decon is not discussed here)
- You measure a seismogram **s** and you have an estimated seismic wavelet **w**, you want to estimate the reflectivity **r**. In this model we also need to consider the presence of noise **n**. We assume Gaussian noise (zero mean) and of variance σ_n^2
- $s_k = (w * r)_k + n_k$
- $\mathbf{s} = \mathbf{W}\mathbf{r} + \mathbf{n}$
- We estimate **r** by minimizing cost function

$$J = \|\mathbf{n}\|_2^2 = \|\mathbf{s} - \mathbf{W}\mathbf{r}\|_2^2$$

Deconvolution

Naive solution $\sigma_n = 0$

$$\hat{\mathbf{r}} = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{s}$$

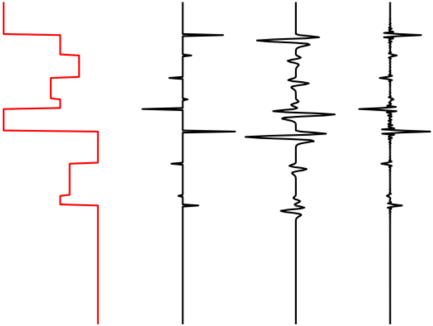


Figure: True impedance and reflectivity, seismogram and estimated reflectivity

Deconvolution

Naive solution $\sigma_n = 0.01$

$$\hat{\mathbf{r}} = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{s}$$

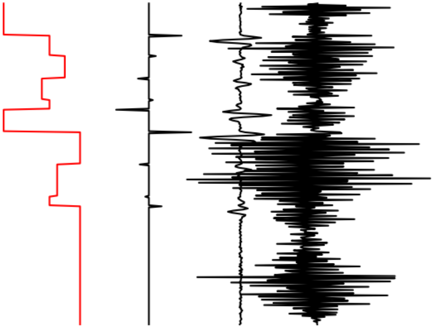


Figure: True impedance and reflectivity, seismogram and estimated reflectivity

Deconvolution and Tikhonov Regularization

Regularized solution $\sigma_n = 0.01$, $\mu = 0.05$.

$$\widehat{\mathbf{r}} = (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{s}$$

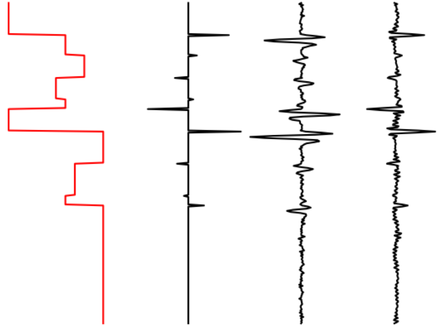


Figure: True impedance and reflectivity, seismogram and estimated reflectivity

Deconvolution and formal derivation of Tikhonov regularization

Tikhonov Regularization ≡ add a stability constraint that prevents the norm of the solution becoming large.

Define new cost function:

$$J(\mathbf{r},\mu) = \|\mathbf{s} - \mathbf{W}\mathbf{r}\|_2^2 + \mu\|\mathbf{r}\|_2^2$$

Solution is given by

$$\widehat{\mathbf{r}}_\mu = \arg\min_{\mathbf{r}} J(\mathbf{r},\mu)$$

$$\widehat{\mathbf{r}}_\mu = (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{s}$$

Solution depends on trade-off parameter $\mu > 0$.

Deconvolution and Tikhonov Regularization

Consider out initial problem $\mathbf{s} = \mathbf{W}\mathbf{r} + \mathbf{n}$ and, let's call the ideal data $\mathbf{s_0} = \mathbf{W}\mathbf{r}$

$$
\begin{aligned}
\hat{\mathbf{r}}_\mu =& (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{s} && (1)\\
=& (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T(\mathbf{s_0} + \mathbf{n}) && (2)\\
=& (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T(\mathbf{W}\mathbf{r} + \mathbf{n}) && (3)\\
=& (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{W}\mathbf{r} + (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{n} && (4)
\end{aligned}
$$

If $\mu = 0$

$$\hat{\mathbf{r}}_\mu = \mathbf{r} + (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{n}$$

But $\mu \neq 0$ (and assuming $(\mathbf{W}^T\mathbf{W})^{-1}$ exists)

$$\hat{\mathbf{r}}_\mu = \mathbf{R}\mathbf{r} + (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{n}$$

where $\mathbf{R}$ is the residual wavelet

$$\mathbf{R} = (\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{W}$$

Deconvolution and Tikhonov Regularization

$$\widehat{\mathbf{r}}_\mu = \underbrace{\mathbf{Rr}}_{1} + \underbrace{(\mathbf{W}^T\mathbf{W} + \mu\mathbf{I})^{-1}\mathbf{W}^T\mathbf{n}}_{2}$$

1. Blurring increases as μ increases
2. Noise increases as μ decreases

Deconvolution and Tikhonov Regularization

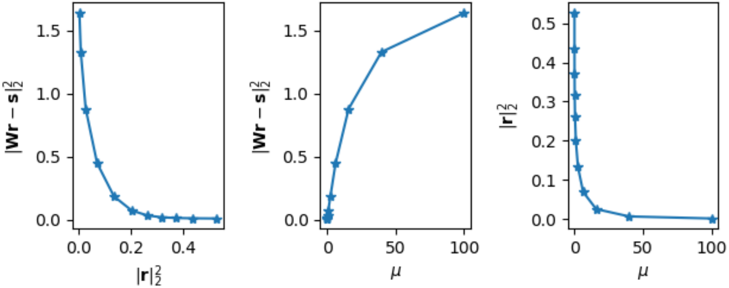


Figure: Tradeoff curve