Tuesday Nov 9

Term paper presentations start Tues Nov 16

- Schedule is on Canvas (including peer review assignments)
  - In ‘Term paper’ Module on Canvas
- Your lectures should be tutorial in nature
- Papers can be more technical

Term papers are due December 11

- ~ 2000 words plus figures and bibliography (at least 4 references)
- Detailed rubric for term paper and presentation on Canvas

Lowest homework score will be dropped (so you can skip HW8 if you wish)

Student talks today –

Andrew

Elize

Matt

Last time – Machine Learning, Christina Kumler

Today – Convolution, Neural networks, Principal Component Analysis

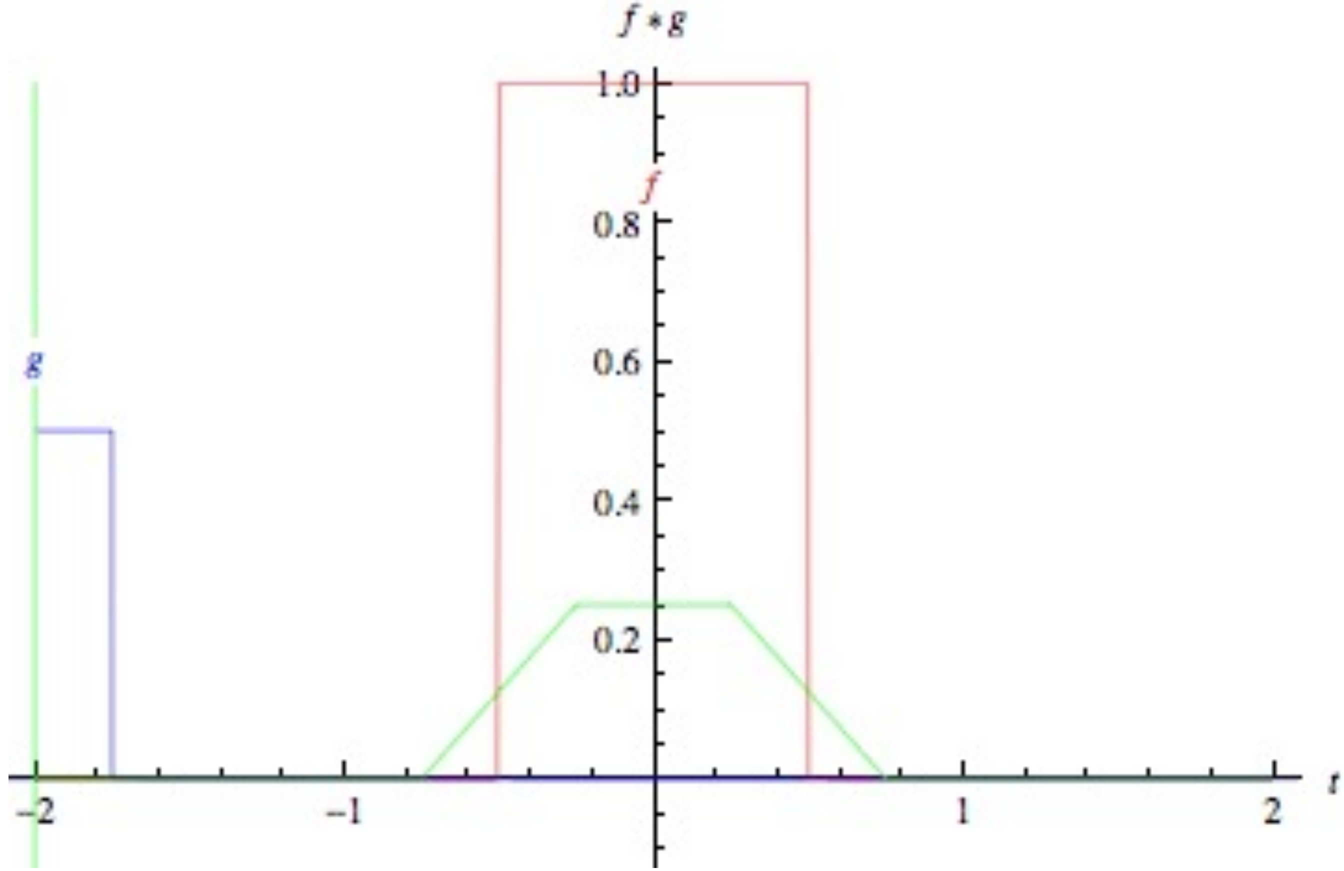
Next time – Term paper presentations start

Convolution

- A function derived from two given functions by integration that expresses how the shape of one is modified by the other
- Change in waveshape as a result of passing through a linear filter

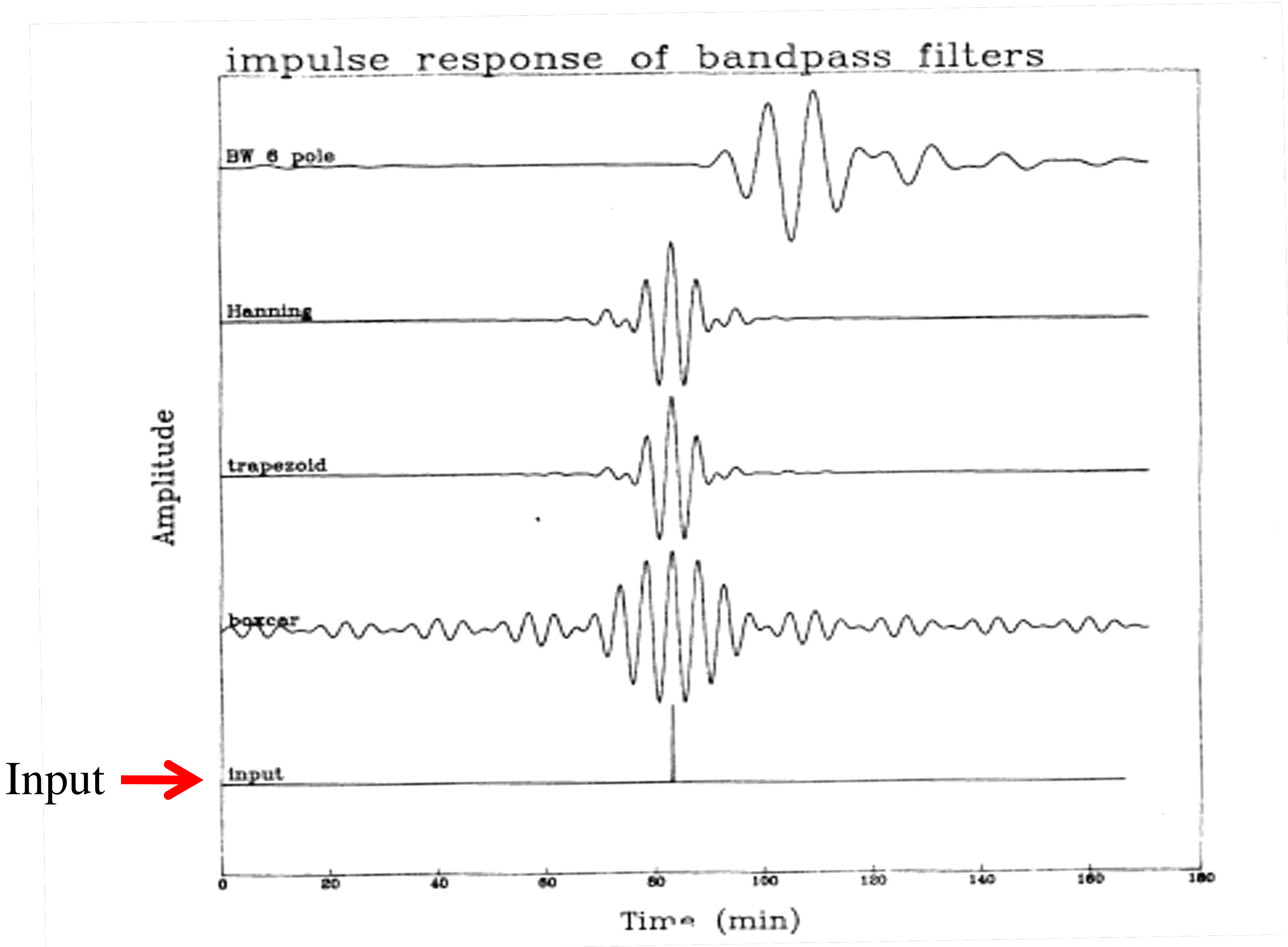
$$f(x) * g(x) \quad = \quad \int_{-\infty}^{\infty} f(\tau) \cdot g(x-\tau)\, d\tau$$

$$f[x] * g[x] \quad = \quad \sum_{k=-\infty}^{\infty} f[k] \cdot g[x-k]$$

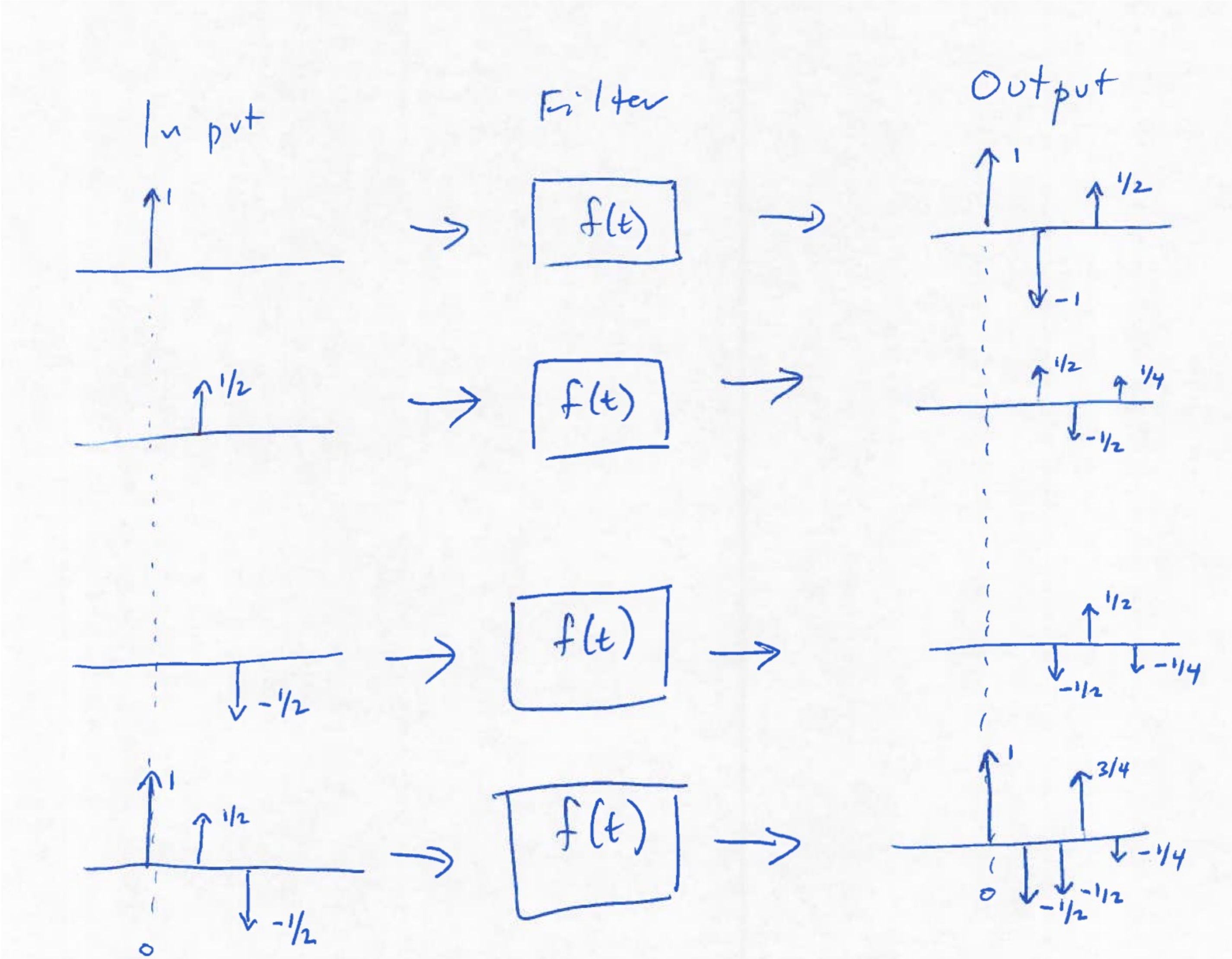


The green curve shows the convolution of the blue and red curves as a function of t, the position indicated by the vertical green line. The gray region indicates the product $g(\tau)f(t-\tau)$ as a function of t, so its area as a function of t is precisely the convolution.

http://mathworld.wolfram.com/Convolution.html


impulse response of bandpass filters
BW 6 pole
Hanning
trapezoid
boxcar
input
Input
Amplitude
0
20
40
60
80
100
120
140
160
180
Time (min)

Convolution



If a waveform g(t) is passed into a linear filter with the impulse response f(t), then the output is given by the convolution of g with f.

Convolution

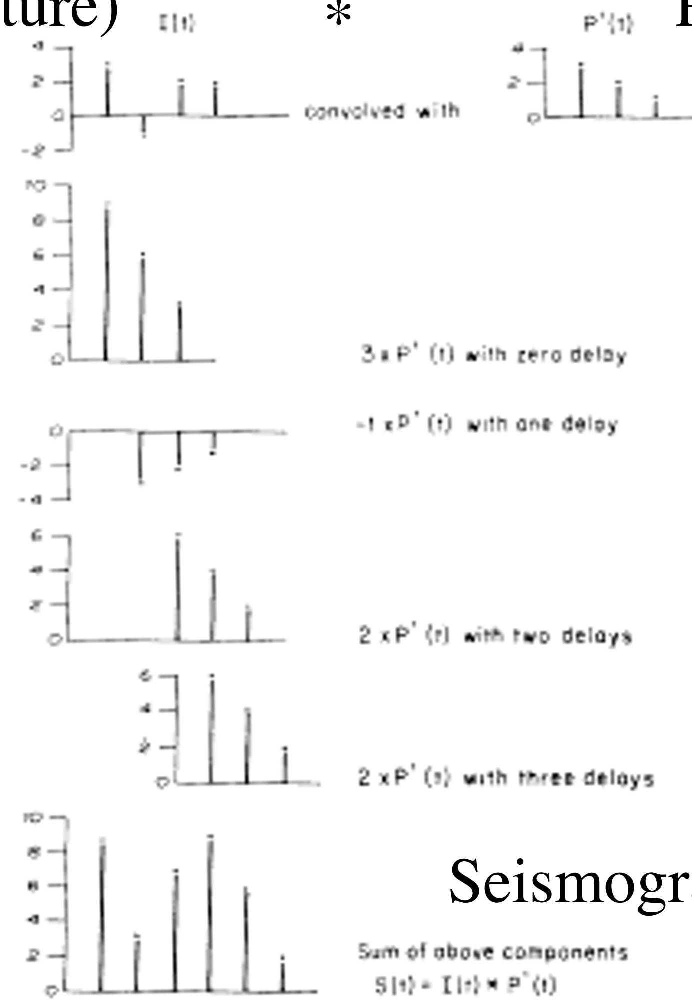
$$S_1 = a_1 p_1$$

$$S_2 = a_1 p_2 + a_2 p_1 \qquad S_i = \sum_{j=1}^{i} a_j p_{i-j+1} \qquad (4.17)$$

$$S_3 = a_1 p_3 + a_2 p_2 + a_3 p_1$$

This process is known as *convolution*, and symbolically we write

A (structure) * P (source)



Seismogram (A*p = S)

FIGURE 4.7 The procedure in a simple convolution.

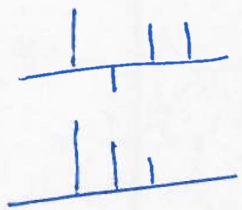
Convolution Example

$$s(t) = A(t) * p(t)$$

$$S_i = \sum_{j=1}^{i} a_j \, p_{i-j+1}$$

$$A = \overset{1}{3}, \overset{2}{-1}, \overset{3}{2}, \overset{4}{2}$$

$$p = \overset{1}{3}, \overset{2}{2}, \overset{3}{1}$$



$$S_1 = a_1 p_1$$

$$S_2 = a_1 p_2 + a_2 p_1$$

$$S_3 = a_1 p_3 + a_2 p_2 + a_3 p_1$$

$$S_4 = \cancel{a_1 p_4} + a_2 p_3 + a_3 p_2 + a_4 p_1 + \cancel{a_5 p_1}$$

$$S_5 = \cancel{a_1 p_5} + \cancel{a_2 p_4} + a_3 p_3 + a_4 p_2 + \cancel{a_5 p_2} + \cancel{a_6 p_1}$$

$$S_6 = \cancel{a_1 p_6} + \cancel{a_2 p_5} + \cancel{a_3 p_4} + a_4 p_3$$

Can set up convolution in matrix form

$$d = G\,m$$

$$\begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \end{bmatrix} = \begin{bmatrix} a_1 & & \\ a_2 & a_1 & \\ a_3 & a_2 & a_1 \\ a_4 & a_3 & a_2 \\ & a_4 & a_3 \\ & & a_4 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

6x1 6x3 3x1

← earth response

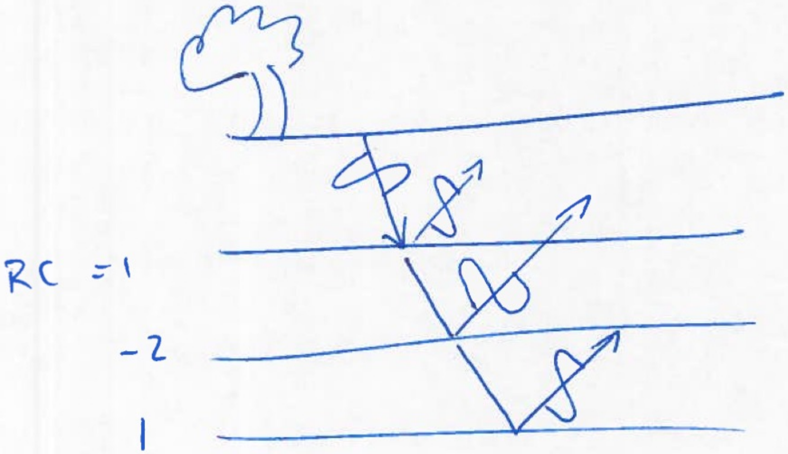
↑ vertical seismogram in convolutional form

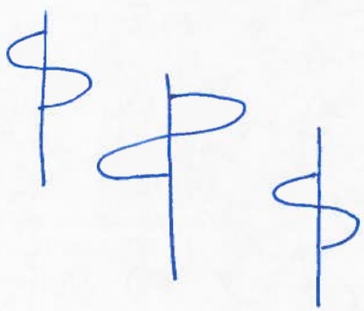
↑ radial seismogram

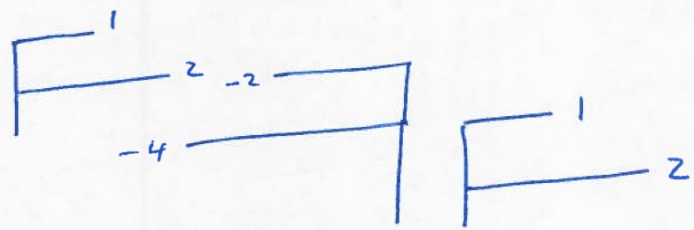
Convolution = forward problem

Deconvolution = inverse problem

Convolution – physical model from seismology



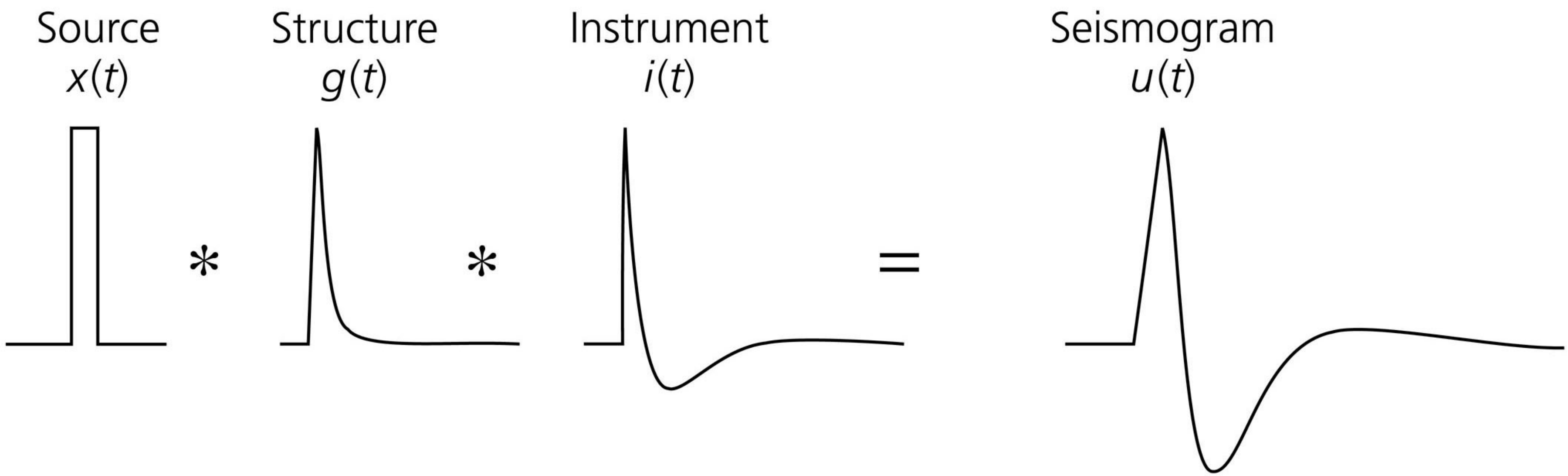




on surface get all reflections together

Convolution
Filtering by the Earth and the Seismometer

Figure 6.3-5: Seismogram as the convolution of the source, structure, and instrument signals.

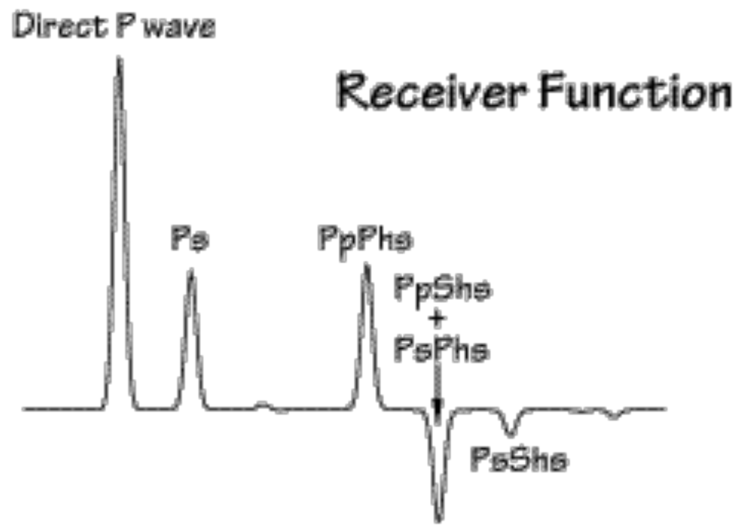


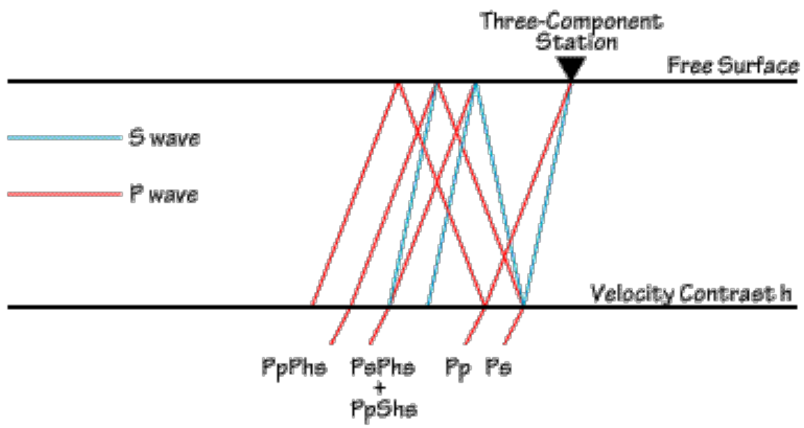
We often want to *deconvolve* source and instrument response to determine the signal from structure

Stein and Wysession

Teleseismic ‘Receiver Functions’ determined by *deconvolving* vertical component recording of teleseism (distant earthquake) from radial component recording at the same receiver (seismic station).

Isolates P-to-S phase conversions at seismic discontinuities (sharp contrasts) in the subsurface.





http://eqseis.geosc.psu.edu/cammon/HTML/RftnDocs/rftn01.html

- Using Fourier analysis, deconvolution is represented by division in the frequency domain (convolution is multiplication in the frequency domain)

- In the time domain, convolution is an integral, and deconvolution is an inverse problem.

Time domain deconvolution

- Time Domain Deconvolution for receiver functions (Sheehan et al., 1995)
  - Band-limited seismic signals make deconvolution using discrete Fourier transforms numerically ill conditioned
  - The Fourier integral for $D_v(t)*E_r(t) = D_r(t)$ (receiver function convolved with vertical = radial) is represented as a discrete sum
  - Stabilization is accomplished with a damping parameter rather than the water level parameter used for spectral deconvolution
  - Solve for m using a damped least squares inversion

$D_v(t)*E_r(t) = D_r(t)$

$$r(t) = v(t) * f(t)$$

$$= \int_{-\infty}^{\infty} v(t-\tau) f(\tau) d\tau \ .$$

Represent as discrete sum

$$r_{k^i} = \sum_{j=0}^{i} v_{k^{i-j}} f^j$$

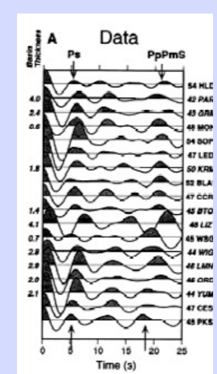
Represent as a system of discrete equations

The G matrix contains the vertical component seismograms

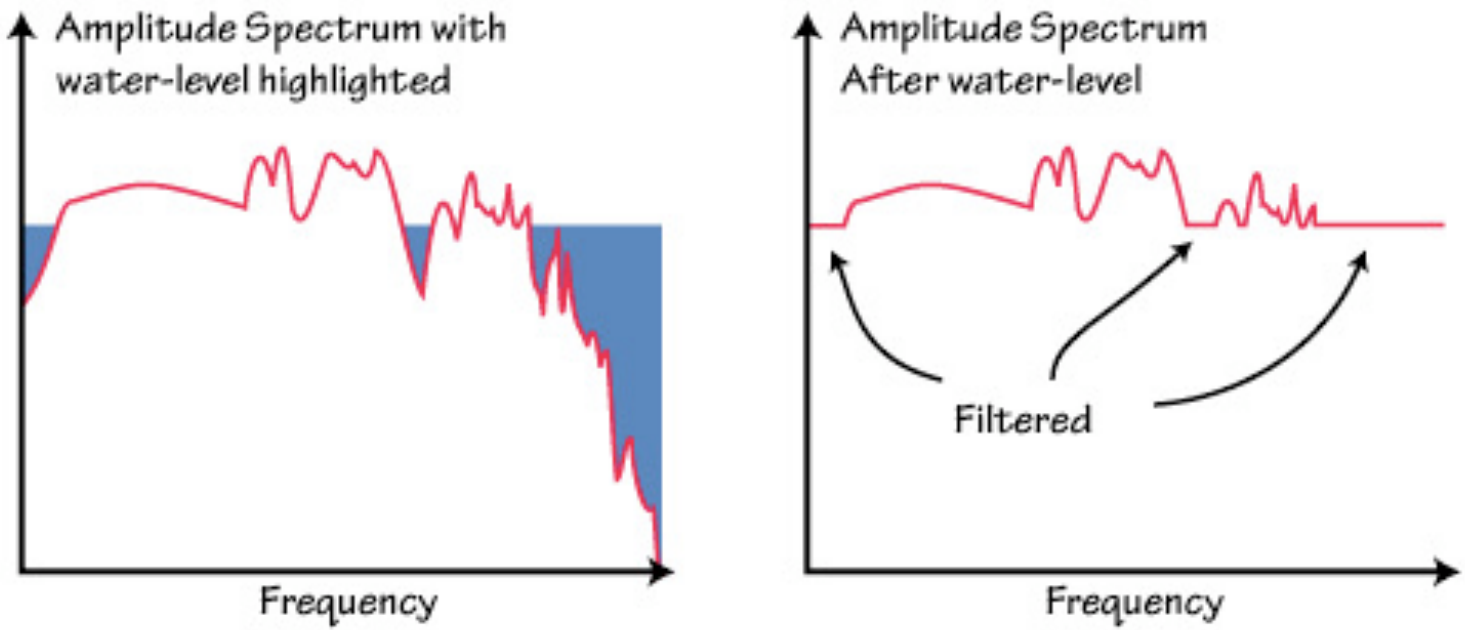
$$\begin{vmatrix} r_1 \\ r_2 \\ \vdots \\ r_K \end{vmatrix} = \begin{vmatrix} G_1 \\ G_2 \\ \vdots \\ G_K \end{vmatrix} \mathbf{f}$$

$$\mathbf{r} = G\mathbf{f}.$$

Solve using damped least squares for f



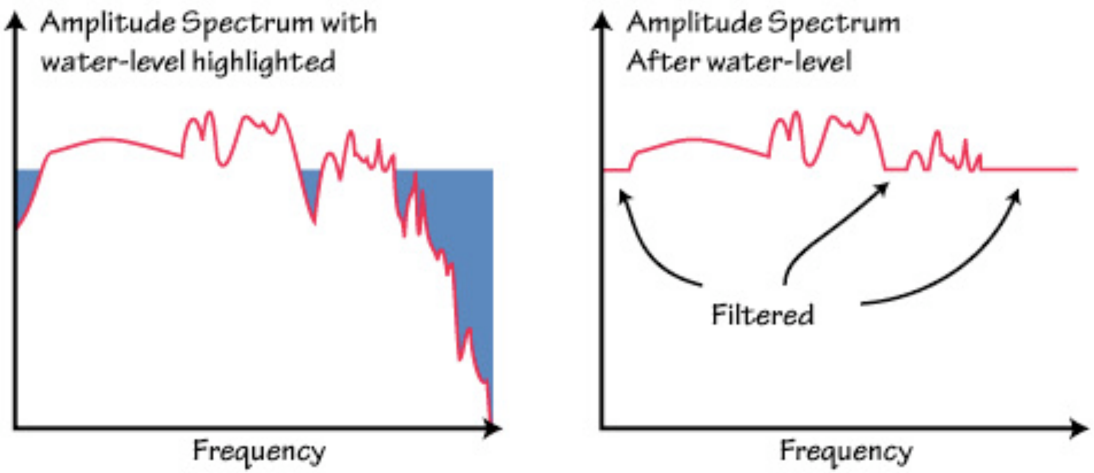
Water level deconvolution in frequency domain



Chuck Ammon's online receiver function tutorial:

http://eqseis.geosc.psu.edu/~cammon/HTML/RftnDocs/rftn01.html

- Stabilize by water level in frequency domain



- Stabilize by damping in time domain

N CRUSTAL THICKNESS VARIATIONS 20,393

$$\begin{vmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \\ \vdots \\ \mathbf{r}_K \end{vmatrix} = \begin{vmatrix} G_1 \\ G_2 \\ \vdots \\ G_K \end{vmatrix} \mathbf{f} \qquad (3)$$

or

$$\mathbf{r} = G\mathbf{f}.$$

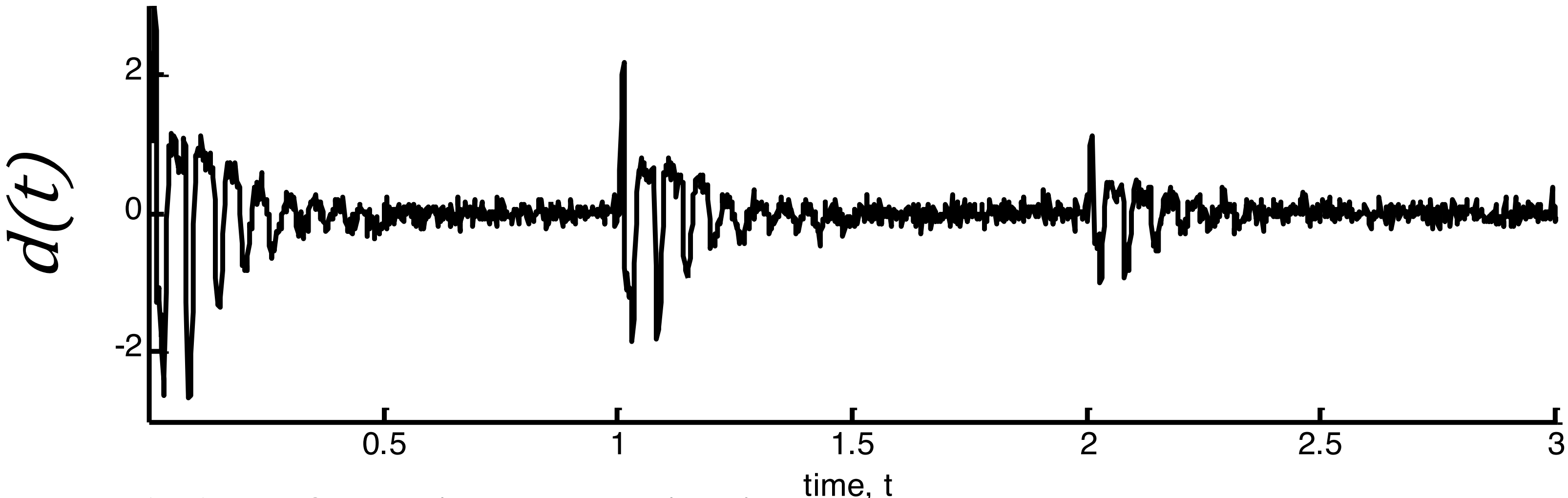
The vector $\mathbf{r}$ has the radial component seismograms for one or more events. If desired, multiple seismograms can go into this vector sequentially, rather than performing "stacks," provided that the seismograms are from events at similar distances and azimuths. The matrix G contains the vertical component seismogram(s) in convolutional (Toeplitz) form

$$G_k = \begin{vmatrix} v_k^0 & 0 & . & 0 \\ v_k^1 & v_k^0 & . & . \\ \vdots & \vdots & \vdots & \vdots \\ v_k^{nk} & v_k^{nk-1} & . & v_k^{nk-m} \end{vmatrix} \qquad (4)$$

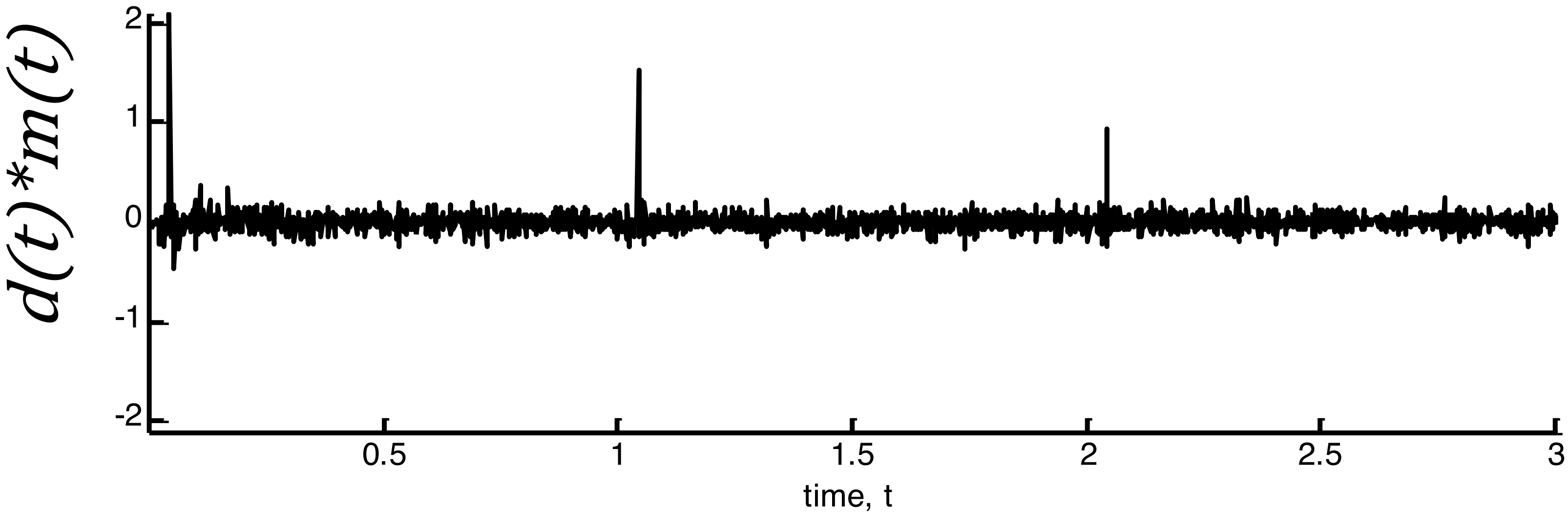
Equation (3) is solved using a damped least squares inversion for $\mathbf{f}$.

Airgun Example (Menke Ch 12)

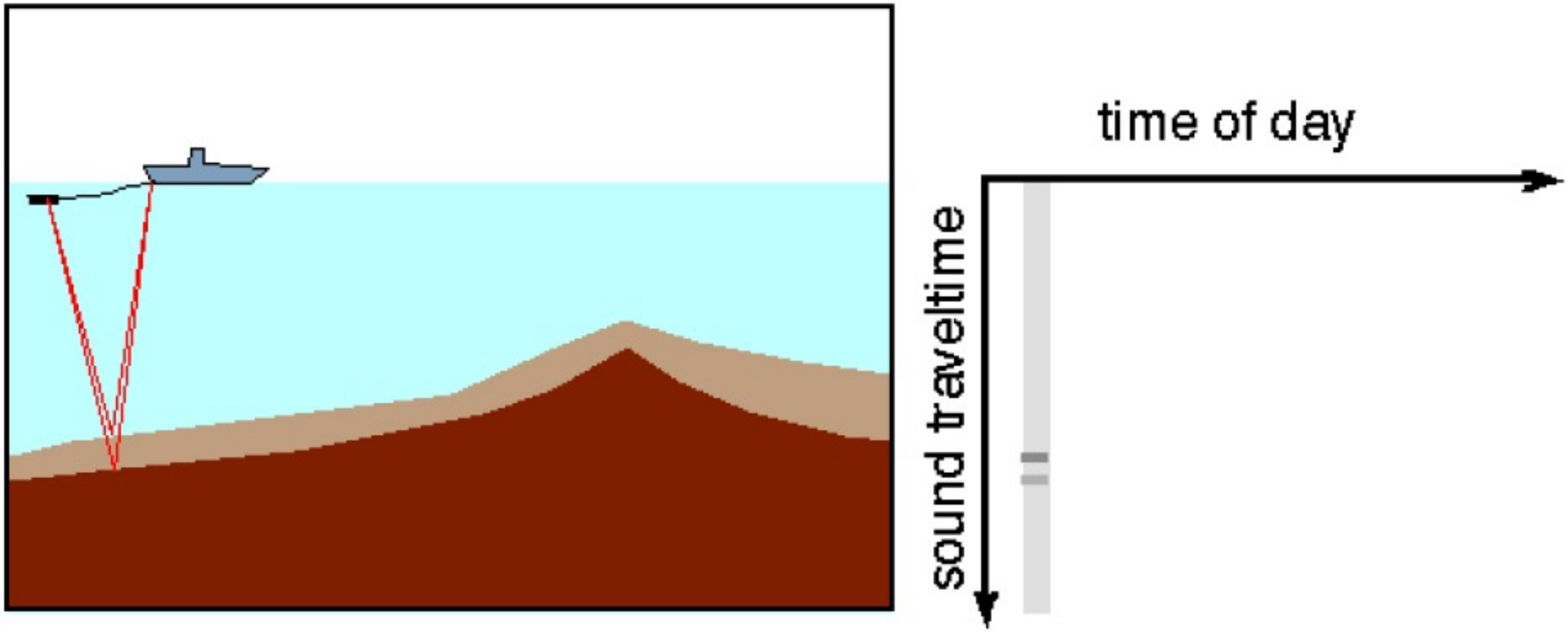
(A) Original

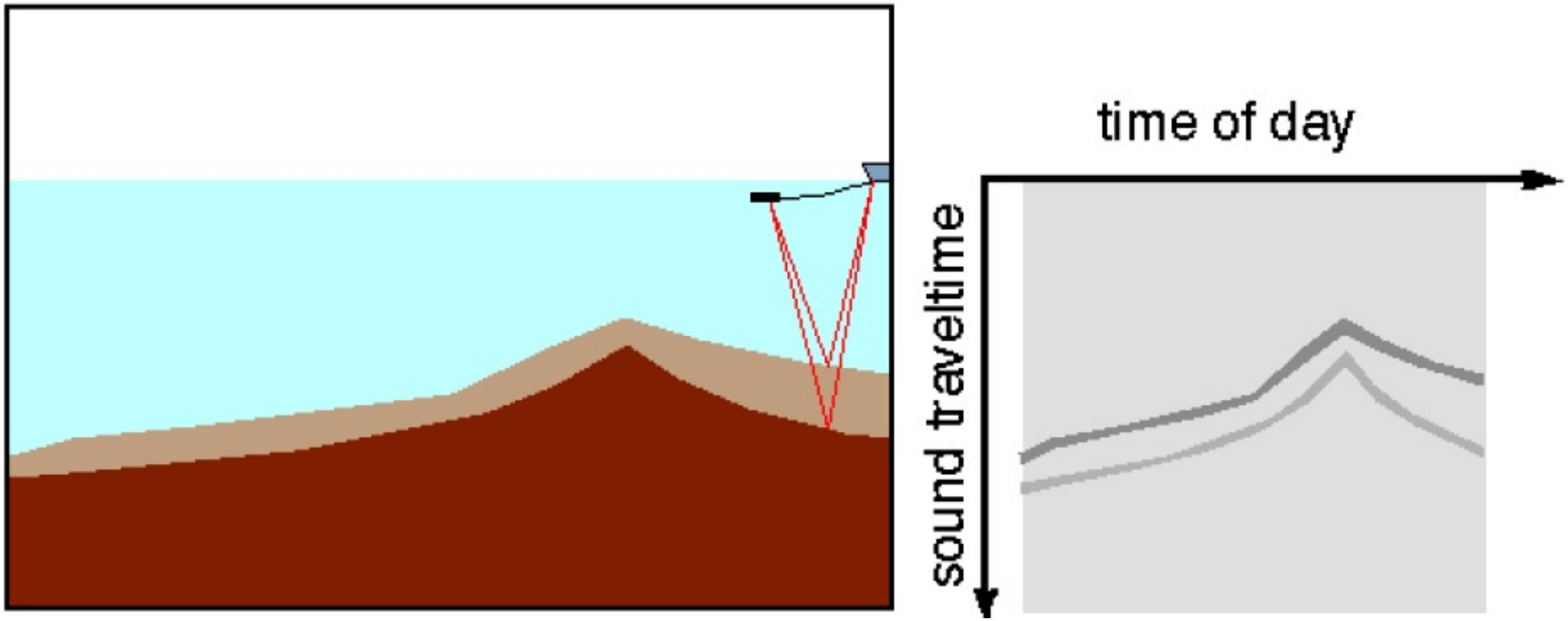


(B) After deconvolution

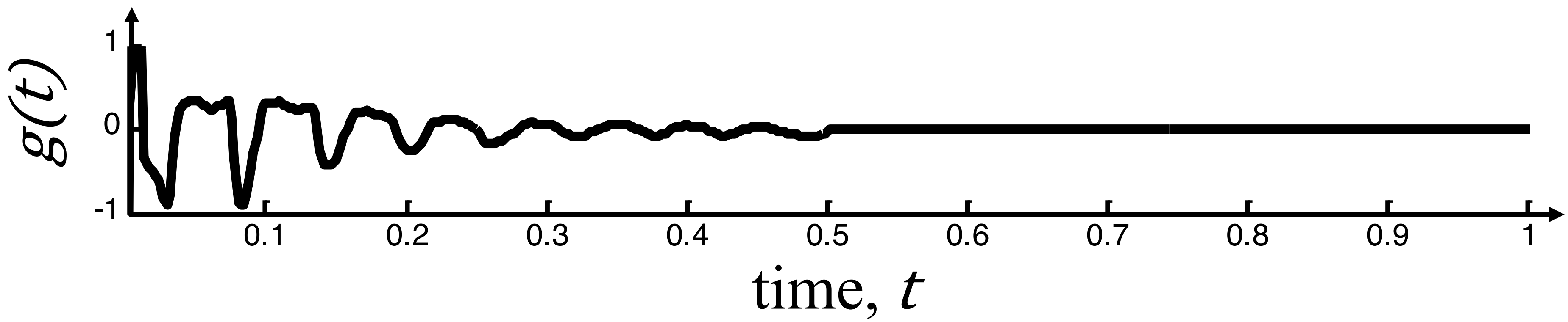


seismic reflection sounding




time of day
sound traveltime

actual airgun pulse is ringy



want airgun pulse to be as spiky as possible

$$p(t) \quad = \quad g(t) \quad * \quad r(t)$$

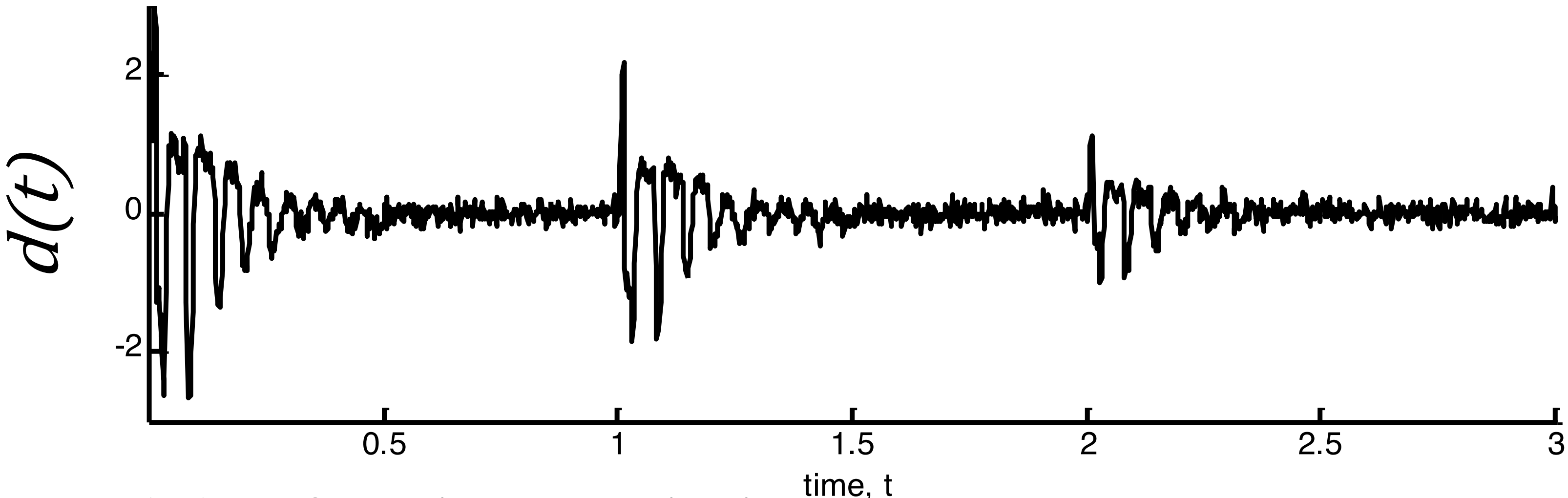
pressure = airgun pulse * sea floor response

so as to be able to detect pulses in sea floor response

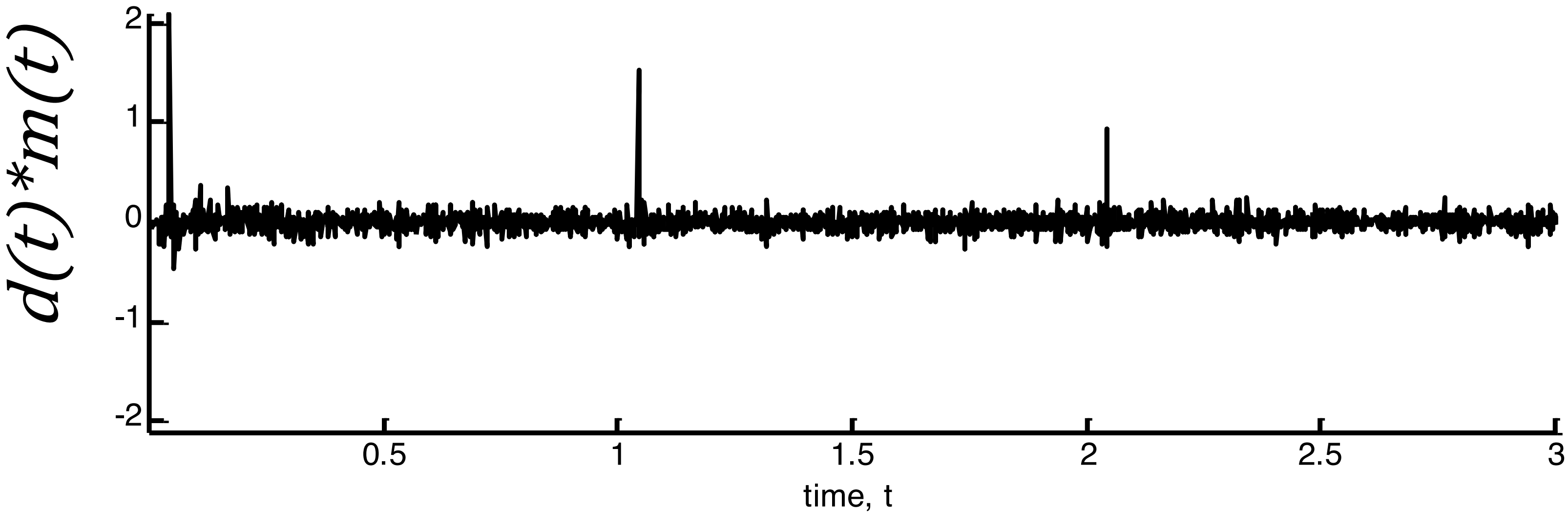
$$p(t) \approx r(t)$$

Airgun Example (Menke Ch 12)

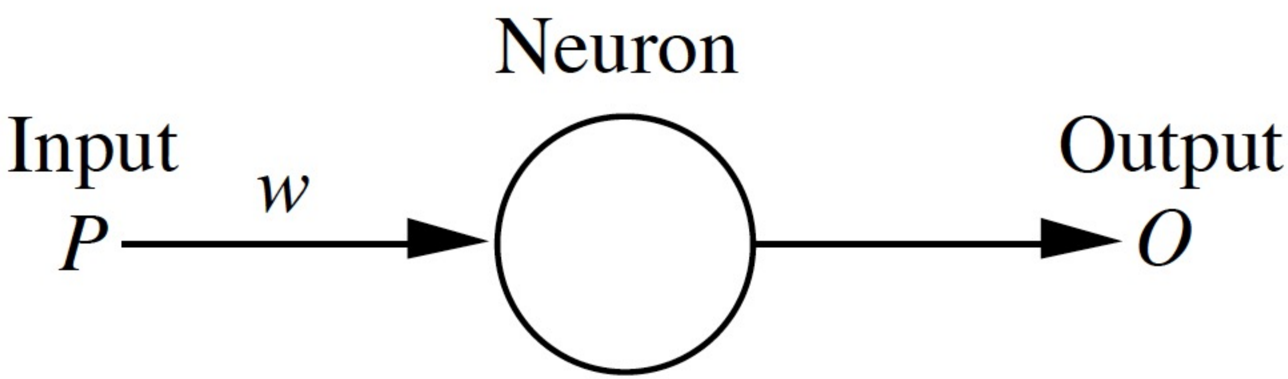
(A) Original



(B) After deconvolution



Neural Network



So, how is the neuron model useful? It is useful because, like the brain, the neuron can be trained and can learn from experience. What it learns is *w*, the correct weight to give the input so that the output of the neuron matches some desired value.

Richardson and Zandt, 2003

Neural Network

A very simple example

We want to train the neuron to learn the correct w by giving it examples of inputs and outputs that are true.

input =	1	output =	2
	10		20
	–20		–40
	42		84

We start out not knowing what w should be. Let's assume w=0.

For first test with input = 1, output would be 0.

That doesn't fit, so w changed. Let's assume a larger w is tried, this time 0.5. Input is 10, output is 5. Still doesn't fit.

When it cycles through all of the input/output pairs once, that is an epoch. It will eventually settle on w=2. The neural network 'learned' this relationship by adjusting weights until it was able to correctly match a set of input/output pairs.

Richardson and Zandt, 2003

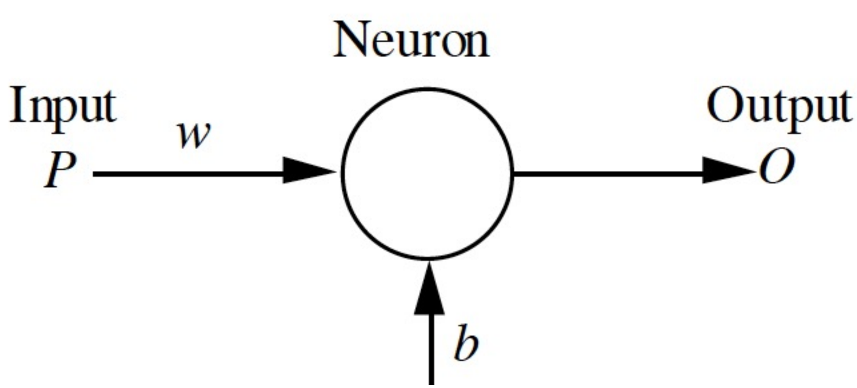
Neural Network

Another simple example

Output shifted 3 units from previous example

input =	1	output =	5
	10		23
	–20		–37
	42		87

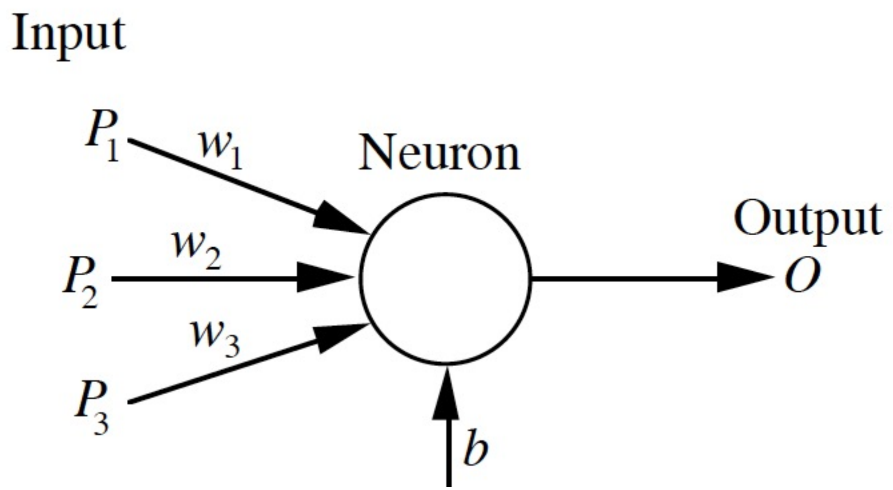
We won't be able to find a single weight w to make the output equal the desired output. Introduce another element, called a bias, to the neuron model. So now b and w will be determined in the learning process.



Richardson and Zandt, 2003

Neural Network

Consider neuron with multiple inputs



Describe the neuron by the function Pw + b, where

$$\mathbf{P} \cdot \mathbf{w} = P_1 w_1 + P_2 w_2 + \cdots + P_N w_N = \sum_{i=1}^{N} P_i w_i$$

Richardson and Zandt, 2003

Factor analysis (also called Principal Component Analysis or Empirical Orthogonal Functions)   (Menke Chapter 10)

Seeks small number of factors that explain much of a data set

Way to reduce the complexity of a model

Example of EOF – finding optimal locations for tsunami observation sites

Mulia et al., GRL, 2017

AGU PUBLICATIONS

Geophysical Research Letters

RESEARCH LETTER
10.1002/2017GL075791

Key Points:
- Empirical orthogonal functions (EOFs) are used to identify effective locations of offshore tsunami measurements
- An optimization method based on a mesh adaptive direct search (MADS) is used to reduce the measurement points
- Smaller number (23 points) of better distributed locations than the existing network would produce slip distributions with smaller errors

Optimal Design for Placements of Tsunami Observing Systems to Accurately Characterize the Inducing Earthquake

Iyan E. Mulia[1], Aditya Riadi Gusman[1], and Kenji Satake[1]

[1]Earthquake Research Institute, University of Tokyo, Tokyo, Japan

Abstract Recently, there are numerous tsunami observation networks deployed in several major tsunamigenic regions. However, guidance on where to optimally place the measurement devices is limited. This study presents a methodological approach to select strategic observation locations for the purpose of tsunami source characterizations, particularly in terms of the fault slip distribution. Initially, we identify favorable locations and determine the initial number of observations. These locations are selected

Tsunami amplitudes at locations x_i ($i=1,\ldots p$)

Snapshots of tsunami amplitudes in time t_j ($j=1,\ldots n$)

Form matrix F with size of n x p

snapshots at time t as column vector

Calculate $R = F^T F$

Solve eigenvalue problem $RC = C\Lambda$

Λ is a diagonal matrix with eigenvalues λ_i of R with corresponding eigenvectors c_i which are column vectors of C

The eigenvectors are the EOF spatial modes, with the first mode associated with the largest eigenvalue. The extrema (min/max) of the first few leading EOF spatial modes are good locations for observation points.

Mulia et al., 2017

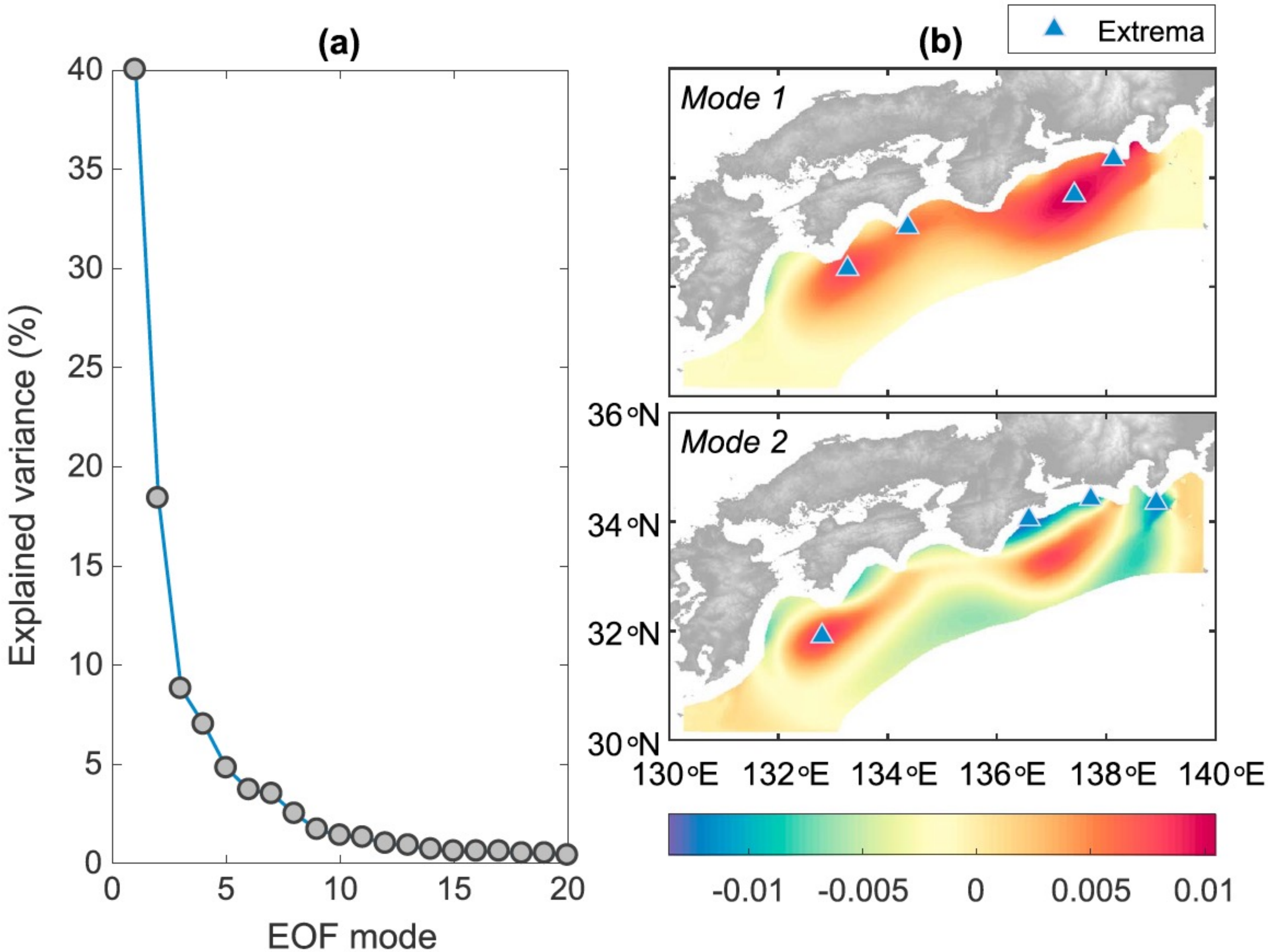


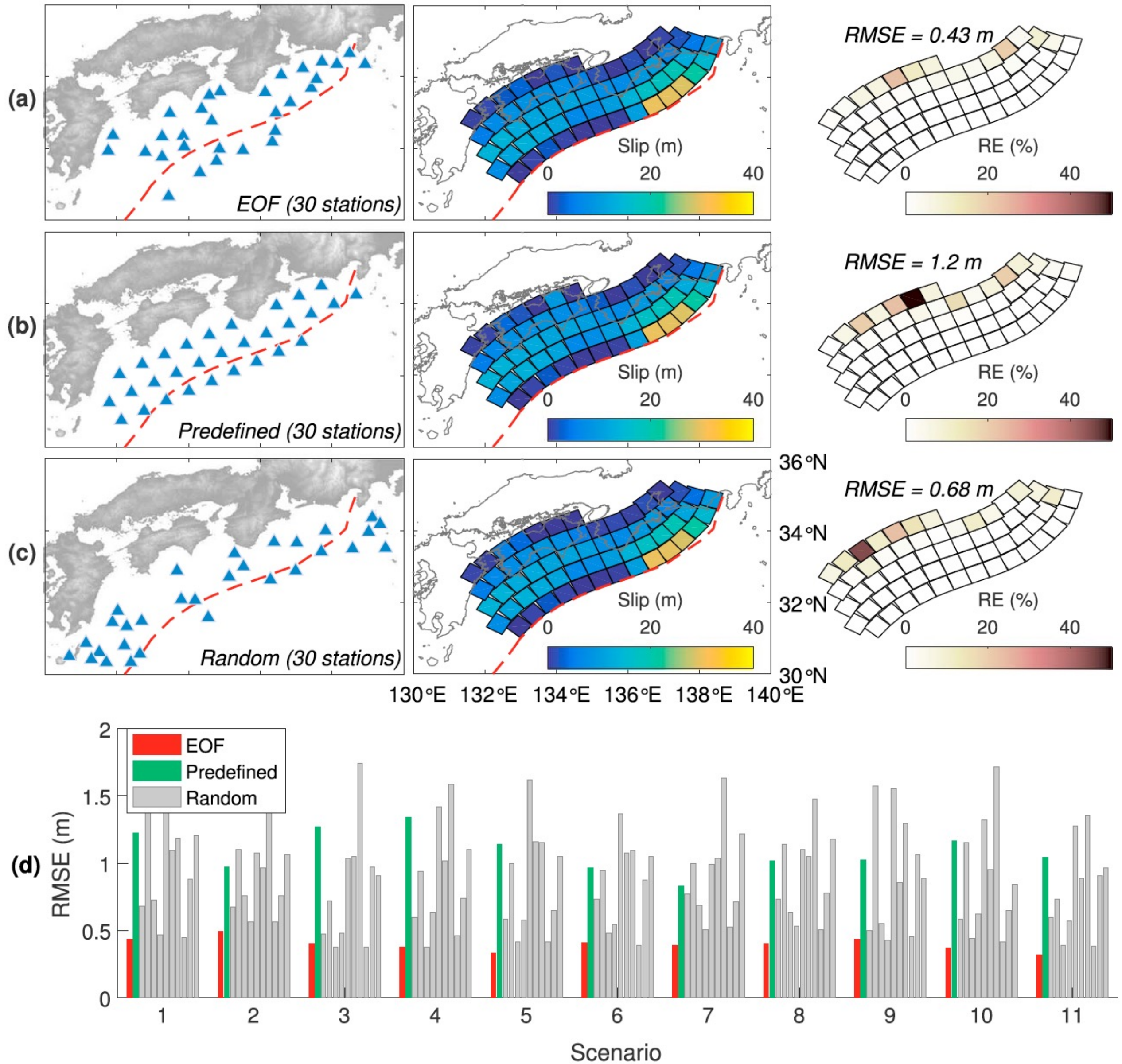
Figure 2. EOF analysis for the scenario 1 (results of the scenario 2 to 11 are shown in the supporting information). (a) Percentage of variance explained by the EOF modes for tsunami amplitudes. (b) The upper and lower figures show contour of the first and second EOF spatial modes (dimensionless), respectively, overlaid with their respective extrema locations.

Mulia et al., 2017

EOF

Uniform grid

Random



RMSE = square root of the mean of the squares of the errors

Figure 3. Performance comparisons between the EOF-generated, predefined, and randomly distributed observation points. (a) EOF-generated points. (b) Predefined points. (c) Random points (one seed). (d) Comparisons of RMSE for all scenarios between the EOF-generated, predefined, and 10 random point seeds. Result examples using the scenario 1 are shown in Figures 3a–3c: station locations (left), inverted slip (middle), and errors (right).

Mulia et al., 2017

- EOF analysis uses a set of orthogonal functions (EOFs) to represent a time series in the following way:

$$Z(x, y, t) = \sum_{k=1}^{N} PC_k(t).EOF_k(x, y)$$

- Z(x,y,t) is the original time series as a function of time (t) and space (x, y).
- EOF(x, y) show the spatial structures (x, y) of the major factors that can account for the temporal variations of Z.
- PC(t) is the principal component that tells you how the amplitude of each EOF varies with time.