Signal Processing and Inverse Theory

Mike Warner

MSc Petroleum Geophysics

Spring 2009
Signal Processing and Inverse Theory

Aims

1. Introduce the principal concepts and methods that are used to analyze, filter and deconvolve simple 1D digital signals.

2. Introduce the general methods that can be used to extract useful geophysical information from incomplete, inconsistent and inaccurate datasets.

Objectives

At the end of the course, you should:

- have a quantitative understanding of simple time-series analysis, convolution, discrete Fourier transforms and their applications, linear filters, deconvolution, and Wiener filters.

- understand the principles of generalized inversion and their application to simple over-determined, under-determined, and non-linear geophysical systems.

Useful Reading

Geophysical Signal Analysis. Enders Robinson and Sven Treitel. Prentice-Hall, 1980. A classic book and a solid introduction to the first half of the course. [Reprinted as a monograph by SEG, not currently in print.]


Seismic Data Processing. Les Hatton, Michael Worthington and John Makin. Blackwell Science, 1986. A condensed but informative account of both sections of the course, and also contains useful material on seismic data processing. [Out of print.]

Time Series Analysis and Inverse Theory. David Gubbins. Cambridge University Press, 2004. A clear account of both parts of the course, but focused on earthquake seismology and geomagnetism rather than seismics – be aware that there are misprints in several places.

Information-Based Inversion and Processing with Applications. Tad Ulrych and Mauricio Sacchi. Elsevier, 2005. Covers both parts of the course, has a useful introduction to both topics, but rapidly moves to a higher level in a broader subject area. [An early version of parts of this text are available at saig.phys.ualberta.ca/book, but watch out for the misprints.]
Content

Signal Processing and Analysis

- Continuous and discrete functions – time series
- Convolution, models of convolution
- $z$-transforms, convolution in the $z$-domain
- Deconvolution
- Dipoles, minimum, maximum and mixed phase
- Fourier transforms, discrete Fourier transforms, aliasing
- Relationship between Fourier and $z$-transforms
- Convolution in the frequency-domain
- Truncation, windowing, tapering
- Properties of Fourier transforms
- Deconvolution in the frequency-domain, stabilisation
- Auto-correlation, cross-correlation
- Correlation in the $z$-domain and as a matrix equation
- Significance of phase

Inverse problems

- Inaccurate, inconsistent and inadequate data
- Matrix equations, norms
- The linear problem
- Equi-determined systems
- Least-squares inversion of over-determined systems, weighting
- The under-determined problem, the null-space, inexact data
- Least-squares inversion of a dipole
- Wiener filters – spiking, shaping, predicting
- Damping and regularisation
- Non-linear systems, linearised inversion
Continuous and discrete functions – time series

Many simple measuring devices generate an output signal that varies continuously with time. For example, a simple geophone generates a continuous voltage related to the motion of the ground. Such a signal can be written as a continuous function of time, for example as \( g(t) \). If a portion of such a signal, lasting from time \( t = 0 \) to \( t = T \), is sampled at regular time intervals \( \Delta t \), then the resulting sequence of \( N \) numbers, \( (g_0, g_1, g_2, \ldots g_n, \ldots g_{N-1}) = \{g\} \) is called a *time series*, where

\[
\begin{align*}
g_n &= g(n \Delta t) \\
n &= 0 \text{ to } N - 1 \\
T &= (N - 1) \Delta t
\end{align*}
\]

The time series \( \{g\} \) is a discrete representation of the original continuous function \( g(t) \). Discrete signals are often described as digital signals, and continuous signals are often described as analogue. Strictly a digital signal will also have a finite precision and finite dynamic range whereas a discrete signal, at least in principle, may not.

A “time” series need not be defined in time; thus the parameter \( t \) could for example represent depth down a well or distance along the Earth’s surface. \( \Delta t \) would then represent regular increments of depth (perhaps representing the positions where discrete measurements were made), or regular intervals along the surface (perhaps representing the positions of geophones or seismic sources). In such cases the resultant sequence is still often referred to as a time series even though time is not involved. Time series are typically formed of a sequence of real numbers. However a regularly spaced sequence of complex numbers also forms a time series; complex series play an important role in time series analysis.

A time series may in principle contain a finite or an infinite number of samples. Commonly the first sample in a time series will represent zero time, but this need not be the case. When the zero time in a time series is significant, and when it does not correspond to the first sample, then it must be specified. One way in which this may be displayed graphically is by use of an arrow at the zero-time sample:

\[
\begin{array}{c}
... \quad a_{-2}, \quad a_{-1}, \quad a_0, \quad a_1, \quad a_2, \quad ...
\end{array}
\]

↑

In expressions involving finite-length time series, the values of samples corresponding to times before the series begins and to times after it ends, are normally considered to be zero. Thus we may write any finite time series as an infinite time series extended at both ends by an infinity of zeros. For example, a time series starting at \( t = 0 \) and containing \( N + 1 \) samples, is equivalent to the infinite time series

\[
\begin{array}{c}
... \quad 0, \quad 0, \quad a_0, \quad a_1, \quad a_2, \quad ... \quad a_N, \quad 0, \quad 0, \quad ...
\end{array}
\]

↑

A time series that only contains zeros before zero time is called a causal time series. Later we will consider time series that represent the action of linear filters on other time series. If such a filter represents the action of a real device or a real process within the Earth, then it must be represented by a causal time series. That is, the output of real devices and real processes can only depend directly upon things that have already happened, and not upon things that will happen in the future.
Convolution

There are many ways in which two functions \(a(t)\) and \(b(t)\), or two time series \(\{a\}\) and \(\{b\}\), may be usefully combined to form a new function \(c(t)\) or a new time series \(\{c\}\). One of the most important of these is convolution written as \(a(t) * b(t) = c(t)\) or \(\{a\} * \{b\} = \{c\}\).

For discrete systems, the convolution \(\{c\} = \{a\} * \{b\}\), where \(\{a\}\) contains \(N\) samples and \(\{b\}\) contains \(M\) samples, is defined by

\[
c_m = \sum_{n=0}^{N-1} a_n b_{m-n}
\]

(1)

where \(m = 0\) to \(M + N - 2\), and the newly formed time series \(\{c\}\) contains \(M + N - 1\) samples.

For continuous signals, the convolution \(c(t) = a(t) * b(t)\) is defined by

\[
c(t) = \int_{-\infty}^{\infty} a(\tau) b(t - \tau) d\tau
\]

where \(\tau\) is a dummy variable that disappears on integration. If \(a(t)\) and \(b(t)\) have finite durations, from \(t = 0\) to \(t = T_A\) and \(t = 0\) to \(t = T_B\) respectively, then this becomes

\[
c(t) = \int_{0}^{T_A} a(\tau) b(t - \tau) d\tau
\]

(2)

and \(c(t)\) has a finite duration from \(t = 0\) to \(t = T_A + T_B\).

Equations (1) and (2) are discrete and continuous expressions of the same relationship. From these definitions, it is straightforward to deduce that

\[a * b = b * a\]

that is, the convolution of \(a\) with \(b\) is identical to the convolution of \(b\) with \(a\); and

\[(aa) * (\beta b) = \alpha \beta (a * b)\]

\[(a_1 + a_2) * b = (a_1 * b) + (a_2 * b)\]

where \(\alpha\) and \(\beta\) are scalar constants; that is, convolution is a linear operation.

Convolution is an operation that arises naturally when considering linear systems and linear filters, be they natural physical systems, man-made devices, or digital computer programs.

If the impulse response of a time-invariant linear system is \(h\) (i.e. the output is \(h\) when the input is a unit impulse at time zero), then the response for some input \(g\) will be simply \(g * h\).

Understanding convolution

The convolution operation may be viewed in several different but equivalent ways; some of these are described below. We will see later that convolution also has a simple form in the frequency and \(z\)-transform domains.

Convolution by scale, delay and stack

Draw or write the longer time series as a vertical column akin to a reflection seismic trace. For each live sample in this trace, scale the other time series by the value of that sample, and
delay the scaled time series by the time of that sample. Do this for every live sample, and then stack the resulting delayed traces to obtain the convolution. This viewpoint is most useful when one time series is significantly longer than the other, and/or contains rather few widely-spaced live samples.

Convolution by reverse and slide

Another method of viewing convolution is to consider it in terms of sliding two time series past each other. This is done by reversing one of the series in time, and then sliding one past the other while multiplying overlapping values and adding the results to give the output. This is best illustrated by example.

If one time series is $(1, 2, -2)$ and the other is $(1, 0, -2)$, then reversing the second and overlaying it with the first gives

\[
\begin{array}{ccc}
1 & 2 & -2 \\
-2 & 0 & 1 \\
\end{array}
\]

Multiplying the elements of each column together gives

\[
\begin{array}{cccc}
0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 \\
\end{array}
\]

which when added together gives 1. The value of the convolution at time zero is therefore 1.

Similarly, to obtain the next output sample, slide the reversed series by one sample and repeat the process. This gives

\[
\begin{array}{ccc}
1 & 2 & -2 \\
-2 & 0 & 1 \\
\end{array}
\]

which when multiplied along columns and added together gives 2.

Repeating this for all shifts until one series has completely passed by the other gives the final result $(1, 2, -4, -4, 4)$. This viewpoint most closely matches the definition.

Convolution by matrix multiplication

The convolution $\{a\} \ast \{b\} = \{c\}$ may be written as a matrix equation

\[Ab = c\]
where $\mathbf{b}$ is a column vector of length $M$ that contains the elements of $\{b\}$, $\mathbf{c}$ is a column vector of length $M+N-1$ that contains the elements of $\{c\}$, and $\mathbf{A}$ is a rectangular matrix with $M$ columns and $M+N-1$ rows that contains the elements of $\{a\}$ arranged as

\[
\mathbf{A} = \begin{bmatrix}
a_0 & 0 & \ldots & 0 \\
a_1 & a_0 & 0 & \vdots \\
a_2 & a_1 & \ddots & \vdots \\
a_3 & a_2 & \ddots & 0 \\
\vdots & \vdots & \ddots & a_0 \\
a_{N-1} & a_{N-2} & \cdots & a_1 \\
0 & a_{N-1} & \vdots & \vdots \\
0 & 0 & \ddots & \vdots \\
\vdots & \vdots & \ddots & a_{N-2} \\
0 & 0 & \ldots & a_{N-1}
\end{bmatrix}
\]

Multiplying out the matrix equation generates the convolution.

**Question 1**

Convolve the time series $(1, 0, -1, 2)$ and $(1, -2)$ by each of the three methods above and show that they each produce the same result.

**$z$-transforms**

There are many useful ways to transform functions and time series into other functions or time series. One type of transform that can be applied only to discrete data is the $z$-transform.

The $z$-transform of the time series $\{a\} = (a_0, a_1, a_2, \ldots, a_n, \ldots)$ is defined\(^1\) as a polynomial in the complex variable $z$ using the elements of the time series as coefficients, thus

\[ A(z) = a_0 + a_1 z + a_2 z^2 + \ldots + a_n z^n + \ldots \]

For example, the time series $(1, -3, 4, -1)$ has the $z$-transform

\[ A(z) = 1 - 3z + 4z^2 - z^3 \]

The $z$-transform provides an alternative means of describing a time series.

\[ \{a\} \iff A(z) \]

Both the time series and the $z$-transform representations contain the same information, but the $z$-transform version allows the manipulation of that information using the rules of simple algebra. If we multiply a $z$-transform by the variable $z$, we obtain

\[ z A(z) = a_0 z + a_1 z^2 + a_2 z^3 + \ldots + a_n z^{n+1} + \ldots = B(z) \]

This new $z$-transform $B(z)$ corresponds to the time series

\[ \{b\} = (0, a_0, a_1, a_2, \ldots, a_n, \ldots) \]

\(^1\)In electrical engineering, the $z$-transform is defined with $z^{-1}$ replacing $z$ throughout. Among other effects, this change moves the zeros of a minimum-phase signal from outside to inside the unit circle.
which is just the original time series delayed by one sample; that is, it is delayed by a time \( \Delta t \). Thus we can regard the variable \( z \) as a unit delay operator that delays a time series by one time step.

If a time series contains elements before time zero, then these are used as coefficients for negative powers of \( z \), thus the time series
\[
1, \quad -2, \quad 3, \quad 5, \quad -1
\]
has the \( z \)-transform
\[
A(z) = z^{-2} - 2z^{-1} + 3 + 5z - z^2
\]

**Convolution in the \( z \)-domain**

Consider the convolution of two time series \( \{a\} \) and \( \{b\} \) and the result \( \{c\} = \{a\} \ast \{b\} \). For example
\[
\begin{align*}
\{a\} &= (a_0, a_1, a_2, a_3, a_4) \\
\{b\} &= (b_0, b_1, b_2) \\
\{c\} &= \{a\} \ast \{b\} = (c_0, c_1, c_2, c_3, c_4, c_5, c_6)
\end{align*}
\]
where
\[
\begin{align*}
c_0 &= a_0b_0 \\
c_1 &= a_1b_0 + a_0b_1 \\
c_2 &= a_2b_0 + a_1b_1 + a_0b_2 \\
c_3 &= a_3b_0 + a_2b_1 + a_1b_2 \\
c_4 &= a_4b_0 + a_3b_1 + a_2b_2 \\
c_5 &= a_4b_1 + a_3b_2 \\
c_6 &= a_4b_2
\end{align*}
\]
Now, \( z \)-transform \( \{a\} \) to \( A(z) \) and \( \{b\ \) to \( B(z) \)
\[
A(z) = a_0 + a_1z + a_2z^2 + a_3z^3 + a_4z^4
\]
\[
B(z) = b_0 + b_1z + b_2z^2
\]
and multiply the two \( z \)-transforms together to obtain
\[
A(z) \times B(z) = a_0b_0 + (a_1b_0 + a_0b_1)z + (a_2b_0 + a_1b_1 + a_0b_2)z^2 + (a_3b_0 + a_2b_1 + a_1b_2)z^3 + (a_4b_0 + a_3b_1 + a_2b_2)z^4 + (a_4b_1 + a_3b_2)z^5 + a_4b_2z^6
\]
This is just the \( z \)-transform \( C(z) \) of \( \{c\} = \{a\} \ast \{b\} \). Thus multiplying the \( z \)-transforms of two time series is equivalent to convolving together the original time series.
\[
\{a\} \ast \{b\} \iff A(z) \times B(z)
\]
We will see later that Fourier transforms have the same property which suggests an intimate relationship between the \( z \) and frequency domains.
Deconvolution

Having defined what it means to convolve two time series \( \{ a \} \) and \( \{ b \} \) together to get \( \{ c \} \), we might ask whether, given the two time series \( \{ a \} \) and \( \{ c \} \), might we be able to reverse the convolution and so recover \( \{ b \} \)? Under appropriate circumstances, this deconvolution operation is possible, but it is not always as straightforward as it might at first appear.

One possible approach to deconvolution would be to start from equation (1), and try to solve for the terms \( b_0, b_1, \ldots \) by some form of recursion – that is, find \( b_0 \) first, then use that to find \( b_1 \), then use those results to find \( b_2 \), etc. The equation to do this is

\[
b_m = \left( c_m - \sum_{n=1}^{m} a_n b_{m-n} \right) / a_0
\]

where \( m \) begins at 0, and equation (4) is applied repeatedly, increasing \( m \) by 1 each time to whatever value is required to recover \( \{ b \} \) fully.

**Question 2**

If \( \{ c \} = (1, 5, 14, 27, 34, 17, 14) \), and \( \{ a \} = (1, 2, 3, 1, 2) \), use the recursive equation (4) to show that \( \{ b \} = (1, 3, 5, 7, 0, 0, \ldots) \). Finally, convolve \( \{ a \} \) and \( \{ b \} \) together to obtain the original \( \{ c \} \).

In this example, the deconvolution is straightforward, and the result is accurate.

**Question 3**

If \( \{ c \} = (1, -1, 0) \), and \( \{ a \} = (1, 1, -2) \), use the recursive equation (4) to obtain the first five terms of \( \{ b \} \). As before, convolve \( \{ a \} \) and \( \{ b \} \) together to try to obtain the original \( \{ c \} \).

In this example, although \( \{ c \} \) and \( \{ a \} \) have only a small number of terms, there is no finite-length result for \( \{ b \} \), and the coefficients of \( \{ b \} \) increase without limit.

If the first term in \( \{ a \} \) is zero, then the recursion in equation (4) cannot even begin because it is not valid to divide by \( a_0 = 0 \). In this case, assuming that \( c_0 \neq 0 \), the deconvolved time series for \( \{ b \} \) must start before time zero.

In many real deconvolution problems, there will be additional complications:

- The result of the convolution \( \{ c \} \) may contain noise so that the problem to be solved is to find \( \{ b \} \) given that \( \{ a \} * \{ b \} + \{ \epsilon \} = \{ c \} \) where \( \{ a \} \) and \( \{ c \} \) are known, and all that is known about the noise \( \{ \epsilon \} \) are some of its statistical properties but not its detailed form.

- The exact form of \( \{ a \} \) may not be known. In this case all that we may have to work with might be an estimate of \( \{ a \} \) or some knowledge of its statistical properties.

- A practical consideration will apply when the time series get very long, and/or when there are very many of them to be solved. In this case, the computational efficiency of deconvolution may become important.

In such cases, the simple recursive deconvolution above will not provide a useful solution, and we will need to use more appropriate techniques for practical deconvolution.
Complex numbers

A complex number $z$ can be written in a cartesian or polar notation (left-hand diagram below), or as a complex exponential. All three are equivalent.

$$z = x + iy = r (\cos \theta + i \sin \theta) = re^{i\theta}$$

**Question 4**

The central figure above shows the unit circle and a complex number $z$ plotted in the complex plane. On this diagram, plot the approximate position of:

(a) $-z$

(b) $z^*$ (see footnote\(^2\))

(c) $z^{-1}$

(d) $z^2$

(e) $\sqrt{z}$ (there are two answers)

On the complex plane on the right-hand figure, plot the position of:

(a) $\sqrt{(-1)}$ (there are two answers)

(b) $\sqrt[3]{(-1)}$ (there are three answers)

(c) $\exp\left\{\frac{2n\pi}{5}\right\}$ where $n$ is an integer (there are five answers)

(d) the two solutions to the quadratic equation $(1 + 0.75z)(1 - 1.25iz) = 0$

**Dipoles**

A simple means of visualizing the properties of a time series in the $z$-domain is by factorising the polynomial into elementary functions or dipoles of the form

$$1 + az$$

where $a$ is a number that may be complex.

\(^2\)The complex conjugate $z^*$ is written as $\bar{z}$ in some texts.
For example, the $z$-transform of $\{x\} = (4, 12, -1, -3)$ can be factorised to give

$$X(z) = 4 + 12z - z^2 - 3z^3 = 4 \left(1 + \frac{1}{2}z\right) \left(1 - \frac{1}{2}z\right) (1 + 3z)$$

Since multiplication of $z$-transforms is equivalent to convolving the original time series, we can also express this as

$$\{x\} = (4, 12, -1, -3) = 4 \left(1, \frac{1}{2}\right) \ast \left(1, -\frac{1}{2}\right) \ast (1, 3)$$

So if we can understand the properties of a single elementary dipole, together with the operation of convolution, then we can extend this to any arbitrary time series by convolving dipoles together.

**Minimum phase dipoles**

Let us assume a dipole $D(z) = 1 + az$ where $|a| < 1$. This dipole corresponds to a time series with two elements $(1, a)$. Now assume that we want to compute an inverse filter for this dipole. That is, we want to compute a new function $F(z)$ such that

$$D(z) \times F(z) = 1$$

Thus

$$F(z) = \frac{1}{D(z)} = \frac{1}{1 + az}$$

Now we can expand this as a binomial series to give the geometric series

$$F(z) = 1 - az + (az)^2 - (az)^3 + (az)^4\ldots$$

Since $|a| < 1$, this series converges.

Now $F(z)$ is just the $z$-transform of the time series

$$\{f\} = (1, -a, a^2, -a^3, a^4\ldots)$$

which represents the inverse of the original dipole $\{d\}$. The convolution of the original dipole with this inverse filter should just give a 1 at zero time

$$\{d\} \ast \{f\} = (1, a) \ast (1, -a, a^2, -a^3, a^4\ldots) = (1, 0, 0, 0, 0,\ldots)$$

which represents a single spike at zero time.

A *minimum-phase* dipole is defined as a dipole $(\alpha, \beta)$ for which $|\alpha| > |\beta|$. We have shown that such a dipole has a causal inverse given by $(1/\alpha)(1, -a, a^2, -a^3, a^4\ldots)$ where $a = \alpha/\beta$ so that $|a| < 1$.

If $|a| \approx 1 < 1$, then the coefficients of the inverse filter will only slowly tend to zero. If on the other hand $|a| \approx 0$, then only a few coefficients will be required to properly represent the inverse of the dipole.

**Question 5**

Compute the first five terms of the inverses of the two dipoles $(1, 0.9)$ and $(1, 0.1)$.

It is clear that when $a = 0.1$, we can truncate the expression adequately after only a few terms, but when $a = 0.9$, we cannot.

To show the significance of such a truncation, convolve the first five terms of the inverses with their respective dipoles, and test how well this approximates $(1, 0, 0, \ldots)$.  

11
It is clear that the truncation error is negligible when \( a = 0.1 \), but that this is not true when \( a = 0.9 \). This problem can be overcome by adopting a least-squares strategy to compute the inverse filter – this is the basis for Wiener filtering.

Another way to describe a minimum-phase dipole is that it is a dipole that has a \( z \)-transform that has a root of the polynomial, say \( z = z_0 \), that lies outside the unit circle in the complex plane\(^3\).

The roots of a polynomial are often called the zeros of the polynomial since the polynomial has the value zero when \( z = z_0 \). For the dipole \( \{a\} = (1, a) \) with \( z \)-transform \( A(z) = 1 + az \), \( A(z) = 0 \) when \( z = z_0 = -1/a \), and since \( |a| < 1 \), then \( |z_0| > 1 \), and so the root will lie outside the unit circle.

**Maximum phase dipoles**

Elementary signals of the form \( (\alpha, \beta) \) where \( |\beta| > |\alpha| \) are called maximum phase dipoles. A maximum-phase dipole has a zero inside the unit circle.

The inverse of a maximum-phase dipole is a non-causal sequence. If the maximum-phase dipole \( (1, b) \) has a \( z \)-transform \( D(z) \), and an inverse \( F(z) \), then

\[
F(z) = \frac{1}{D(z)} = \frac{1}{1 + bz} = 1 - bz + (bz)^2 - (bz)^3 \ldots
\]

Now, since \( |b| > 1 \), this is clearly not a convergent series, and the magnitude of the coefficients \( (1, -b, b^2, -b^3 \ldots) \) will increase with time without limit.

We can however still calculate a useful inverse as follows

\[
F(z) = \frac{1}{1 + bz} = \frac{1}{bz (1 + (bz)^{-1})}
\]

This expression does have a stable expansion of the form

\[
F(z) = (bz)^{-1} \left( 1 - (bz)^{-1} + (bz)^{-2} - (bz)^{-3} \ldots \right)
\]

Now the inverse is stable but non-causal. The inverse filter \( \{f\} \) is given by

\[
\{f\} = \ldots, -b^{-3}, b^{-2}, -b^{-1}, 0 \uparrow
\]

\(^3\)If the electrical engineering definition of the \( z \)-transform is used, then the zeros of a minimum-phase dipole lie inside the unit circle.
Minimum-phase, maximum-phase and mixed-phase signals

A seismic signal is more complicated than a simple dipole. But we can always factorise the $z$-transform of the signal in terms of elementary dipoles (and a multiplicative constant). That is, for a time series $\{x\}$, we have

$$X(z) = x_0 + x_1z + x_2z^2 + x_3z^3 + \cdots + x_nz^n = x_0(1 + a_1z)(1 + a_2z)(1 + a_3z)\cdots(1 + a_nz)$$

Now:

1. if $\{x\}$ is a minimum-phase signal, then $|a_n| < 1$ for all $n$;
2. if $\{x\}$ is a maximum-phase signal, then $|a_n| > 1$ for all $n$;
3. if neither 1 nor 2 is true, then $\{x\}$ is a mixed-phase signal.

A minimum-phase signal has all the zeros of its $z$-transform outside the unit circle, and a maximum-phase signal has all the zeros of its $z$-transform inside the unit circle.

Fourier Transforms

Recall that the complex form of the Fourier series for a continuous periodic signal $g(t)$ with period $T$ is given by

$$g(t) = \sum_{n=-\infty}^{n=+\infty} c_n \exp \left\{ \frac{2\pi int}{T} \right\}$$

where

$$c_n = \frac{1}{T} \int g(t) \exp \left\{ -\frac{2\pi int}{T} \right\} dt$$

and the integration is over any complete period $T$.

If $f(t)$ is a non-periodic function that coincides with $g(t)$ over some interval $T$, then the Fourier series above will also represent the non-periodic function within that interval. Outside this interval, the Fourier series will not represent $f(t)$; it will simply repeat the description with a period of $T$. 
A Fourier series describes a function in terms of the amplitude and phase of its spectral components where the frequency of each component is \( n/T \) and the circular frequency \( \omega \) of each component is \( 2\pi n/T \). The spectral component that corresponds to \( n = 0 \) is termed the DC or zero frequency component – it is equal to the average value of the function.

Recall also that the Fourier transform\(^4\) of a continuous function \( g(t) \) is given by

\[
G(\omega) = \int_{-\infty}^{\infty} g(t) e^{-i\omega t} \, dt \tag{4}
\]

and

\[
g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(\omega) e^{i\omega t} \, d\omega \tag{5}
\]

Both the descriptions of the Fourier series and the Fourier transform above apply to continuous signals, but we can extend these concepts to define the Fourier transform of a discrete time series. The result is another discrete series in which each sample represents a different frequency rather than a different time.

Suppose that we have a time series \( \{x\} = (x_0, x_1, \ldots, x_n, \ldots, x_{N-1}) \), then we can transform this into a new series \( \{X\} = (X_0, X_1, \ldots, X_m, \ldots, X_{N-1}) \) where each sample now represents a value at a particular frequency. The elements of \( \{X\} \) are given by

\[
X_m = \sum_{n=0}^{N-1} x_n \exp\left\{ -\frac{2\pi inm}{N} \right\} \tag{5}
\]

and the elements of the original time series \( \{x\} \) can be recovered from \( \{X\} \) by\(^5\)

\[
x_n = \frac{1}{N} \sum_{m=0}^{N-1} X_m \exp\left\{ \frac{2\pi inm}{N} \right\} \tag{6}
\]

These equations define the forward and inverse discrete Fourier transforms (DFT). They are equivalent to the Fourier transforms defined for continuous signals, and are analogous to the Fourier series representation of a non-periodic function over a single period.

Fourier transforms, whether continuous or discrete, provide two ways of representing the same information: either as a function/series in time or as function/series in frequency. Both representations are equivalent and are equally valid, and the definitions above provide mathematical recipes that allow the derivation of one from the other.

Although we have used time \( t \) and frequency \( \omega \) as the variables in this discussion, we could equally well have used distance \( d \) and wavenumber \( \kappa \) (= \( 2\pi \)/wavelength), or indeed any two variables that have a similar complementary relationship. The Fourier transform of a function of time is often called its frequency spectrum; the Fourier transform of a function of position is often called its wavenumber spectrum.

\(^4\)Fourier transforms may be written in several forms. The negative exponent may be in either the forward or inverse transform, and the factor of \( 1/2\pi \) may appear in the forward or inverse transform, or be distributed between both as \( \sqrt{1/2\pi} \). If frequency rather than circular frequency is used to define the transform, then there is no factor of \( 1/2\pi \) outside the integral, but a \( 2\pi \) appears within both exponents. In geophysics, it is common, but by no means universal, to use circular frequency \( \omega \), to use the negative exponent in the forward transform, and to place the factor of \( 1/2\pi \) in the inverse transform. Any of the above descriptions may be used provided that they are maintained consistently throughout a problem.

\(^5\)The discrete Fourier transform can also be written with the factor of \( 1/N \) in the forward transform rather than the backward transform, or it can be distributed as a factor of \( \sqrt{1/N} \) in both transforms. The minus sign in the exponential can also move between the forward and backward transforms.
In general the Fourier transform \( G(\omega) \) of a function \( g(t) \) is complex, as are the components \( X_m \) of the Fourier transform of a time series \( \{x\} \), so that

\[
G(\omega) = R(\omega) + iI(\omega) = A(\omega) \exp\{i\Phi(\omega)\}
\]

and

\[
X_m = R_m + iI_m = A_m \exp\{i\Phi_m\}
\]

The function \( A(\omega) = |G(\omega)| \) and the values \( A_m = |X_m| \) are called the amplitude spectra of \( g(t) \) and \( \{x\} \). The square of the amplitude spectrum is called the power spectrum or the power spectral density (PSD).

The function \( \Phi(\omega) = \tan^{-1}(I(\omega)/R(\omega)) \) and the values \( \Phi_m = \tan^{-1}(I_m/R_m) \) are called the phase spectra of \( g(t) \) and \( \{x\} \). The values of the phase spectrum always have an associated \( 2n\pi \) ambiguity. Phase spectra are often plotted unwrapped; that is, the \( 2\pi \) ambiguity is partially removed by plotting the spectrum without discontinuous jumps of \( 2\pi \) occurring. Although the unwrapped spectrum is smooth, there is still a \( 2n\pi \) ambiguity associated with each value.

**Aliasing**

When a continuous signal is sampled to produce a discrete time series, there is a maximum frequency, the Nyquist frequency, in the original continuous signal that can be properly captured. The Nyquist frequency depends upon the sample interval, and is given by

\[
f_N = \frac{1}{2\Delta t} \quad \text{or} \quad \omega_N = \frac{\pi}{\Delta t}
\]

Note that the Nyquist frequency is half the sampling frequency of \( 1/\Delta t \). For the common seismic sample interval of 4 ms, the Nyquist frequency is 125 Hz.
If the original continuous signal contains frequencies above the Nyquist frequency, then these frequencies are wrapped around to appear as spurious lower frequencies after discrete sampling. If the Nyquist frequency is \( f_N \), then signals with an original frequency of \( f_N + f \) will appear after discretisation with a frequency of \( f_N - f \).

Frequencies above the Nyquist must therefore be removed by analogue low-pass filtering before discretisation to produce the time series. For data in time, this is straightforward using analogue electrical anti-alias filters; however, such filters will introduce distortion into both the amplitude and phase spectra at frequencies below but close to the Nyquist frequency. In seismics, anti-alias filters are commonly set to roll off at a frequency that is between a half and three-quarters of the Nyquist.

For data that are discretely sampled in space, for example by placing sources or receivers at discrete intervals, spatial aliasing can also occur. This is exactly analogous to temporal aliasing, and it leads to wrap around of spurious wavenumbers. It must be overcome by the design of the field experiment before the wavefield is sampled at discrete intervals. Typically this is achieved by using arrays of closely-spaced geophones or hydrophones that are combined together electrically to form a single signal, and by using arrays of airguns, explosive shots or vibrator positions for the sources.

These acquisition source-and-receiver arrays serve as low-pass spatial anti-alias filters, removing or at least suppressing energy that has a high wavenumber (short wavelength) across the array. Suppressing spatial aliasing in this way is often only partly successful in real surveys, and can be particularly difficult to achieve in the cross-line direction during 3D acquisition. During processing, data are often sorted and/or sub-sampled spatially, for example by CDP sorting, and these processes should be undertaken having proper regard to the potential for re-introducing spatial aliasing.
Relationship between the DFT and $z$-transform

The $z$-transform of $\{x\}$ is given by

$$X(z) = \sum_{n=0}^{N-1} x_n z^n$$

and the DFT of $\{x\}$ is given by

$$X_m = \sum_{n=0}^{N-1} x_n \exp\left\{\frac{-2\pi imn}{N}\right\}$$

Comparing these two equations, it is clear that the DFT is just a $z$-transform in which $z$ has the special value

$$z = \exp\left\{\frac{-2\pi im}{N}\right\}$$

for each component $X_m$.

We can think of the $z$-transform therefore as related to the time domain (each coefficient in the polynomial in $z$ corresponds to a time sample $x_n$) or as related to the frequency domain (if $z$ takes the value $\exp\{-2\pi im/N\}$, then the value of the polynomial at that point corresponds to the frequency sample $X_m$). In some sense therefore, we might consider the $z$-transform as sitting partway between the time domain and the frequency domain.

We can plot the values of $z$ that correspond to the DFT in the complex plane. Since $|\exp\{-2\pi im/N\}| = 1$, these values all lie on the unit circle, and are equally spaced around it from 0 to $2\pi$. The component corresponding to $m = 0$ is the DC component, and the component corresponding to $m = N/2$ is at the Nyquist frequency.

The components below the real axis in the complex plane correspond to positive frequencies that run from 0 to the Nyquist frequency. The components above the real axis in the complex plane correspond to negative frequencies that also run from 0 to the Nyquist frequency.

Recall that a single-frequency cosine wave can be written as

$$A \cos \omega t = \frac{A}{2} \left(e^{j\omega t} + e^{-j\omega t}\right)$$
It is clear from this expression that a cosine is equivalent to two complex exponentials, representing two vectors in the complex plane, that each varies with time; one represents a counter-clockwise rotating vector and the other a clockwise rotating vector. Together these sum to give a real number that is equal to $A \cos \omega t$. We can regard a cosine therefore as being composed of two components with frequencies $+\omega$ and $-\omega$.

**Understanding the frequency components**

Equation (5) can be used to calculate not only the terms $X_0...X_{N-1}$, but also values for $X_m$ for which $m < 0$ or $m > N - 1$. If we calculate these values, then we find that $X_m = X_{N+m}$. That is, the $X$ generated by the DFT can be thought of as a periodic infinite series that repeats every $N$ samples.

Likewise, equation (6) can be used to calculate values for the time series $x_n$ for which $n < 0$ or $n > N - 1$. In this case, we find that $x_n = x_{N+n}$. That is, in applying a discrete Fourier transform, we are implicitly assuming that the time series is also a periodic infinite series in time that repeats every $N$ samples.

In equation (5), each element of the DFT corresponds to a particular frequency. The first sample $X_0$ is the DC component, and the subsequent samples are spaced $\Delta \omega$ apart in frequency where

$$\Delta \omega = \frac{2\pi}{N \Delta t}$$

if the original time series contains $N$ samples spaced in time at $\Delta t$. Equations (5) and (6), representing the forward and backward DFTs, can therefore also be written as

$$X_m = X(\omega_m) = \sum_{n=0}^{N-1} x(t_n) \exp \{-i\omega_m t_n\}$$  \hspace{1cm} (7)

$$x_n = x(t_n) = \frac{1}{N} \sum_{m=0}^{N-1} X(\omega_m) \exp \{i\omega_m t_n\}$$  \hspace{1cm} (8)

where

$$t_n = n \Delta t$$

$$\omega_m = m \Delta \omega = \frac{2\pi m}{N \Delta t}$$

Equations (7) and (8) are exactly equivalent to equations (5) and (6), but they make more explicit the meaning of the individual terms in $\{X\}$.

<table>
<thead>
<tr>
<th>$X_2$</th>
<th>$X_1$</th>
<th>$X_0$</th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_4$</th>
<th>$X_5$</th>
<th>$X_6$</th>
<th>$X_7$</th>
<th>$X_8$</th>
<th>$X_9$</th>
<th>$X_{10}$</th>
<th>$X_{11}$</th>
<th>$X_{12}$</th>
<th>$X_{13}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2\Delta \omega</td>
<td>-\Delta \omega</td>
<td>0</td>
<td>\Delta \omega</td>
<td>2\Delta \omega</td>
<td>3\Delta \omega</td>
<td>4\Delta \omega</td>
<td>5\Delta \omega</td>
<td>6\Delta \omega</td>
<td>7\Delta \omega</td>
<td>8\Delta \omega</td>
<td>9\Delta \omega</td>
<td>10\Delta \omega</td>
<td>11\Delta \omega</td>
<td>12\Delta \omega</td>
<td>13\Delta \omega</td>
</tr>
</tbody>
</table>

Fourier transform coefficients $\{X\}$ and their corresponding frequencies for $N=12$.

The coefficients and the frequencies repeat every 12 samples. $\omega_N$ is the Nyquist frequency.

Looking at these terms individually, $X_0$ is clearly the frequency component for zero frequency. The next term, $X_1$ is the component for frequency $\Delta \omega$; $X_2$ is the component for frequency $2\Delta \omega$; and $X_n$ is the component for frequency of $n \Delta \omega$. As we progress through the frequency series, the value for $z$ moves clockwise around the unit circle starting from $z = 1$ for the DC
term. Eventually, we will arrive halfway through the series at the term \( X_{N/2} \). This term corresponds to a value of \( z = -1 \), and to a frequency of \( N/(2 \Delta \omega) = \pi/\Delta t \) which is just the Nyquist frequency.

As we move beyond the Nyquist frequency and progress further around the unit circle,

- we can either regard the samples as corresponding to frequencies that are higher than the Nyquist,
- or, more usefully, we can use the relationship \( X_m = X_{N+m} \) to regard these as representing negative frequencies that have absolute values less than the Nyquist.

As we continue around the unit circle, we will eventually come to the last sample \( X_{N-1} \) which we can regard as either having a frequency of \( (N-1) \Delta \omega \) or as \(-\Delta \omega\).

If, as will usually be the case, the original time series \( \{x\} \) only contains real values, then the Fourier coefficients at negative frequencies are simply the complex conjugate of the coefficients at positive frequencies – so that the negative frequencies contain no additional information. That is

\[
X_{-n} = X_n^* 
\]

In this case, the amplitude spectrum at negative frequencies is identical to the amplitude spectrum at positive frequencies; and the phase spectrum is inverted. For a real time series, the DC component is purely real (it is simply the average of the time series, so it must be real if the time series is real); and the Nyquist component is also real.

Note that if the real time series \( \{x\} \) contains \( N \) independent real numbers, then the corresponding frequency series \( \{X\} \) will also contain \( N \) real numbers, but these will be arranged as \( N/2 - 1 \) complex numbers plus the two real numbers that correspond to the DC and Nyquist components. Thus both representations contain the same volume of information. In the most general case, \( \{x\} \) is complex, and the positive and negative frequencies then contain independent information.

**Fourier transforms of convolutions**

We saw that the \( z \)-transform of a convolution generates a multiplication. Fourier transforms, whether discrete or continuous, have the same property. That is, if \( X \) is the Fourier transform of \( x \), and \( Y \) is the Fourier transform of \( y \), then the Fourier transform of \( x \ast y \) is simply \( X \times Y \).

\[
\{x\} \ast \{y\} \iff \{X\} \times \{Y\} \\
x(t) \ast y(t) \iff X(\omega) \times Y(\omega)
\]

That is, if we convolve two functions in the time domain, then we multiply their frequency spectra.

The corollary of this, is

\[
\{X\} \ast \{Y\} \iff N \{x\} \times \{y\} \\
X(\omega) \ast Y(\omega) \iff 2\pi x(t) \times y(t)
\]

That is, when we multiply two functions in the time domain we convolve their frequency spectra – but note the factors of \( N \) and \( 2\pi \).

These relationships have many consequences. One practical computational consequence, is that we can often compute a convolution much more quickly by performing a fast Fourier
transform (FFT) on both time series, multiplying their frequency spectra, and performing an inverse transform. If both of the series to be convolved have many samples, then this approach is much faster than convolving them directly in the time domain.

There is however a potential pitfall in this approach. The result of a convolution should be to generate a time series that is longer than either of the original time series. However, the DFT implicitly assumes that these time series are periodic. Thus a convolution computed by multiplying two Fourier transforms together will wrap around in time so that samples that should appear after the last sample of the original series, will actually appear within the earlier part of the output result. Convolution via the DFT is therefore a circular convolution.

This pitfall is easily avoided by padding the original time series with zero values at the end, so that there is sufficient space for the full length of the convolution to be generated without any wrap around. Such padding will always be required when an operation in some domain nominally increase the number of samples obtained after transformation to another domain.

Padding of the time series by adding additional zeros at the beginning and/or end of an existing series is often required in order to use the FFT algorithm which typically requires that the number of samples in the time series to be transformed is exactly equal to a power of two.

Padding the time series leads to finer sampling in frequency – padding in time does not change the Nyquist frequency, but it does generate more frequency samples, consequently $\Delta \omega$ is reduced.

Padding the Fourier transform with zeros, then transforming back to time, will generate a finer sampling in time – that is, it reduces $\Delta t$. This is a simple way to interpolate a time series. Although this operation increases the Nyquist frequency, it will not of course recover high frequencies within data that have already been aliased.

**Truncation, windowing and tapering**

The Fourier transform of a rectangular boxcar function defined by

$$
g(t) = \begin{cases} 1 & -T/2 < t < +T/2 \\ 0 & \text{otherwise} \end{cases}
$$

is

$$
G(\omega) = \int_{-T/2}^{T/2} e^{-i\omega t} dt = \frac{1}{i\omega} \left( e^{i\omega T/2} - e^{-i\omega T/2} \right) = T \left( \frac{\sin(\omega T/2)}{(\omega T/2)} \right) = Tsinc \left( \frac{\omega T}{2} \right)
$$

where the sinc function is defined by $sinc \ x \equiv \frac{\sin x}{x}$.
The boxcar function has “sharp edges”, and its transform, the sinc function, has an oscillatory structure. This is a common feature of the transform of any function that changes abruptly in time; the more abrupt is the change, then the more oscillatory will be the transform.

If a function is truncated in time, then this is equivalent to multiplying it in the time domain by a boxcar function. This, in turn, is equivalent to convolving the frequency spectrum of the un-truncated function with the spectrum of a boxcar function. The result of this is to broaden the spectrum of the original function, and to make it oscillatory in the frequency domain – the resulting spectrum will be smudged and “ringy”.

Since physical time series always have a beginning and an end, their spectra will always be affected by this process. This can be particularly important when large-amplitude signals are artificially and abruptly truncated in time during processing. This phenomenon affects not only data that are truncated in time, but also data that are truncated in space. There are consequently likely to be artifacts associated with the edges of seismic surveys, the beginning and ends of receiver arrays, missing sources, and similar spurious abrupt lateral changes in amplitude.

Analogous effects also occur if we apply a boxcar function in the frequency domain, or indeed if we apply any sort of frequency filter that has sharp edges. Since a multiplication in the frequency domain represents a convolution in the time domain, if we band-pass filter a dataset abruptly in frequency, then we will convolve it in time with the Fourier transform of the corresponding boxcar. This will extend and smudge any temporal structure, and it will introduce spurious oscillations – “ringing” – around any structure in the time domain, reducing resolution and degrading the appearance of the seismic data.

These effects can be ameliorated, but at a cost, by tapering the data smoothly from maximum amplitude to zero over some significant number of samples – the broader and smoother this taper, then the sharper and less oscillatory will be the response in the other domain. In the frequency domain, such tapering means that we cannot usefully use very sharp filters, and in the time domain it means that we should not use very sharp mutes. Indeed almost anything that varies abruptly in time or space or frequency or wavenumber, or in any other (multi-dimensional) transform domain, will have consequences that are likely to be detrimental unless appropriate care is taken.

One consequence of this is that we cannot have data that is highly resolved in both time and in frequency, and in order for data to be sharp in time, it must be broad-band in frequency.

**Properties of Fourier transforms**

Fourier transforms have a number of important properties:

**Linearity**

\[ ag_1(t) + bg_2(t) \iff aG_1(\omega) + bG_2(\omega) \]

This is one of the most important properties, it means that if we add two signals, then we add their Fourier spectra.

**Duality**

\[ g(t) \iff G(\omega) \]

\[ G(t) \iff 2\pi g(-\omega) \]
Reciprocal broadening

\[ g(at) \iff G(\omega/a)/|a| \]

As one function gets broader the other gets narrower. The factor \(1/|a|\) means that the vertical scale also changes.

**Shifting the axis**

\[ g(t - t_0) \iff G(\omega) \exp\{-i\omega t_0\} \]
\[ g(t) \exp\{-it\omega_0\} \iff G(\omega - \omega_0) \]

Multiplication by a complex exponential represents a shift in time or frequency.

**Derivatives**

\[ \frac{d}{dt} g(t) \iff i\omega \ G(\omega) \]
\[ -it \ g(t) \iff (d/d\omega) \ G(\omega) \]
\[ \frac{d^n}{dt^n} g(t) \iff (i\omega)^n \ G(\omega) \]

Differentiating a function multiplies its spectrum by \(i\omega\); that is, it *magnifies high frequencies*, and it shifts the phase by \(\pi/2\).

For example, geophones measure ground velocity whereas accelerometers measure acceleration which is the differential of velocity. Thus we would expect accelerometers to have an enhanced performance over geophones at high frequencies, and a correspondingly weaker response at low frequencies.

Differentiating a second time repeats the process, so that the spectrum is multiplied by \(\omega^2\), and the total phase shift is \(\pi\), that is, the signal is inverted.

**Integrals**

\[ \int g(t) \ dt \iff G(\omega)/i\omega + \text{constant} \]

Integration *magnifies low frequencies*.

We can Fourier transform differential equations as well as functions. Thus the 1D homogeneous acoustic wave equation

\[ \left( \frac{\partial^2}{\partial t^2} \right) p(x, t) = c^2 \left( \frac{\partial^2}{\partial x^2} \right) p(x, t) \]

can be Fourier transformed in time to give

\[ -\omega^2 P(x, \omega) = c^2 \left( \frac{\partial^2}{\partial x^2} \right) P(x, \omega) \]

If the seismic velocity \(c\) is not a function of \(x\), then the wave equation can also be Fourier transformed in space to give

\[ -\omega^2 \mathcal{P}(\kappa, \omega) = -c^2 \kappa^2 \mathcal{P}(\kappa, \omega) \]

or

\[ \omega^2 = c^2 \kappa^2 \]
which is the dispersion relationship for a sound wave in a homogeneous medium. Note that the second Fourier transform cannot be carried out in this simple fashion if the velocity $c$ is a function of position since, in this case, $c(x)$ must also be Fourier transformed.

### Deconvolution in the frequency domain

Since a convolution in the time domain is equivalent to multiplication in the frequency domain, a deconvolution in the time domain will be equivalent to division in the frequency domain.

\[
\text{time} \iff \text{frequency}
\]

\[
a \ast b = c \iff AB = C
\]

\[
a = c \ast (b)^{-1} \iff A = C/B
\]

Thus one simple way to deconvolve one time series by another, is to Fourier transform both, divide one complex frequency spectrum by the other at each frequency, then inverse transform back into time to obtain the result. Typically we will need to pad the input and output time series avoid wrap around caused by the cyclical nature of convolution in the frequency domain.

The equivalence of deconvolution to division in the frequency domain immediately reveals why deconvolution can be impossible, unstable or noisy. Since convolution represents a frequency-by-frequency multiplication, if the amplitude spectrum of $b$ contains a notch at some frequency (i.e. there is no signal in $b$ at that frequency), then $c$ will also contain a notch at the same frequency. Deconvolution of $c$ by $b$ is then equivalent to the division of zero by zero at that frequency, and the result of such an operation is undefined. In other words, the deconvolution will give nonsense at this frequency, and back in the time domain that will have undesirable consequences – it will likely have spurious oscillations at that frequency.

In practice, if both $b$ and $c$ are physical signals (or even if they are signals stored by a computer that has a finite numerical accuracy) then they are unlikely to be identically zero at some frequency. Instead, at notches in the frequency spectrum, both signals will contain random noise – either physical noise generated by the analogue recording instruments or numerical noise generated by the finite numerical accuracy. Thus, instead of dividing zero by zero, we will actually divide a small random number by another small random number, the result of which can be anything.

This problem does not arise as a result simply of deconvolving explicitly in the frequency domain. Rather, it is a fundamental problem of deconvolution in general that is easiest to understand in the frequency domain.

We can stabilize deconvolution by adding a small (but not too small) number to the denominator so that we never divide by something so close to zero that the answer blows up. That is, in the frequency domain, we calculate $A$ using

\[
A = \frac{C}{B + \epsilon}
\]

where $\epsilon$ is much smaller than $B$ except near the notches in the spectrum that we are trying to deal with. Now, when $B$ is large, $A \approx C/B$ to a high degree of accuracy. But when $B$ and $C$ get very small, $A \approx C/\epsilon$ which itself will be small if $\epsilon$ is chosen appropriately.

As written above, this approach assumes that $B$ is a real positive number. In general of
course, $B$ is complex. We can however make the approach general by using instead

$$A = \frac{C B^*}{|B|^2 + \epsilon}$$

where $\epsilon$ is a real positive number. This is now stable for negative and complex values of $B$. This stabilisation process is often know as pre-whitening or as adding white noise. Both descriptions refer to the fact that the intention is add a small amount of signal at frequencies where there is very little – a white signal is one that has the same amplitude at all frequencies. Adding white noise does not mean adding noise to the output; rather it does the reverse, stabilising a process that might otherwise allow noise in the input to appear with arbitrarily large amplitude in the output.

This stabilisation approach is used in many areas of geophysics outside deconvolution – whenever we divide one number by another, or indeed whenever we carry out any inverse operation, we should seek to stabilise the results to guard against the generation of large spurious factors resulting from division by small, noise-dominated numbers. As a simple example, an algorithm that tries to normalise the amplitudes of seismic traces by scaling using their average amplitudes, will blow up if one of the traces has zero amplitude – or more significantly, it will increase the amplitudes spuriously for any traces that have small amplitudes dominated by noise rather than signal.

**Cross-correlation and auto-correlation**

The cross-correlation $\{r(a,b)\} \equiv \{r_{ab}\}$ of two time series $\{a\}$ and $\{b\}$, that each have a zero mean, is defined\(^6\) as

$$r_{ab}(n) = r_n(a, b) = \sum_m a^*_m b_{m+n}$$

(9)

where the summation is over all possible values of $m$, and $n$ can take both positive and negative values. The cross-correlation $r_{ab}(\tau)$ of two continuous functions $a(t)$ and $b(t)$, that have zero mean, is defined as

$$r_{ab}(\tau) = \int_{-\infty}^{\infty} a^*(t) b(t + \tau) \, dt$$

(10)

The variable $\tau$, and the index $n$ when multiplied by the time step $\Delta t$, are called the time lag, and this controls the temporal shift of one function relative to the other. Cross-correlation provides a measure of the degree of similarity between the two functions, or the two time series, as a function of this time lag. The cross-correlation is defined for both positive and negative time lags, and in general the values at $+\tau$ and $-\tau$, or at $+n$ and $-n$, will be unrelated.

Comparing equations (9) and (10) with equations (1) and (2), it is clear that a cross-correlation is simply a convolution of the complex conjugate of one of the functions with the other function reversed in time. Unlike convolution, cross-correlation does not commute, and when the order of the two functions is switched, then the cross-correlation is reversed in time and conjugated. It is common to write a cross-correlation series with the names of the two original functions indicated in brackets in the correct order or as subscripts. With these conventions therefore, we have

$$r_n(a, b) = r^*_n(b, a)$$

\(^6\)In some fields, the auto and cross-correlation are defined with a normalisation factor of $1/N$ or $1/(N-1)$ in front of the summation.
\[ r_{ab}(n) = r_{ba}^*(-n) \]
\[ r_{ab}(\tau) = r_{ba}^*(-\tau) \]

If the two functions or time series are identical, then the result of the correlation is called the auto-correlation function. It is given by

\[ r_{aa}(n) = r_n(a) = \sum_m a_m^* a_{m+n} = \sum_m a_m^* a_{m-n} \]
\[ r_{aa}(\tau) = \int_{-\infty}^{\infty} a^*(t) a(t + \tau) \, dt = \int_{-\infty}^{\infty} a(t) a^*(t - \tau) \, dt \]

If the original series or function is purely real, then the auto-correlation is real and is symmetric in time. It also has a maximum positive value at a time lag of zero – that is, any function correlates with itself most strongly when there is no time shift.

It is common practice to normalise the auto-correlation so that the value at zero lag is equal to one\(^7\). The cross-correlation is also frequently normalised using

\[ r_n(a, b) = \frac{\sum_m a_m^* b_{m+n}}{\left( \sum_m a_m^2 \right)^{1/2} \left( \sum_m b_m^2 \right)^{1/2}} \]

The absolute value of this normalised function never exceeds 1. Perfect correlation corresponds to 1, perfect anti-correlation corresponds to \(-1\), and no correlation whatsoever corresponds to 0.

From here onwards, we will assume that all signals, wavelets and filters are real, have a zero mean, and are normalised.

**Properties of the auto-correlation**

The normalised auto-correlation function of an infinite sequence of perfectly random noise will have a value of one at a time lag of zero, and will be zero at all other times. Random noise is, by definition, uncorrelated. If the noise sequence has a finite duration, then there will be small contributions to the correlation function at non-zero lags.

![4000 ms of random noise 500 ms auto-correlation function](image)

If a time series is periodic with a certain period, then its auto-correlation function will also be periodic with the same period.

\(^7\)In some fields, the un-normalised correlation is called the covariance, and the term correlation is reserved only for the normalised correlation.
If a periodic time series is also noisy, then the underlying periodicity will be easier to observe in the auto-correlation function than in the original time series.

The cross-correlation of two signals that are composed of the same signal displaced in time, plus random noise, will have a peak in amplitude at a lag corresponding to the original time displacement.

The Fourier transform of the auto-correlation function of a real time series is simply the power spectrum, that is the square of the amplitude spectrum. We can therefore regard the auto-correlation function as effectively a representation of the original time series with the effects of phase removed.

**The auto-correlation function and the z-transform**

The $z$-transform of the auto-correlation function of a time series with $z$-transform $X(z)$ is

$$R(z) = X(z) X^*(z^{-1})$$

that is, convolve the $z$-transform of the time series with the $z$-transform of its complex conjugate reversed in time.

The $z$-transform of the minimum-phase dipole $(1, a), |a| < 1$ is given by

$$D_{\text{min}}(z) = 1 + az$$

and its, auto-correlation function is given by

$$R_{\text{min}}(z) = a^* z^{-1} + \left(1 + |a^2|\right) + az$$
Similarly, the $z$-transform of the maximum-phase dipole $(a^*, 1)$ is given by

$$D_{\text{max}}(z) = a^* + z$$

and its auto-correlation function is given by

$$R_{\text{max}}(z) = a^* z^{-1} + (1 + |a^2|) + az$$

This leads to the important conclusion that

$$R_{\text{max}}(z) = R_{\text{min}}(z)$$

in other words, that two different sequences can have the same autocorrelation function.

Further, since the Fourier transform of the auto-correlation function is just the square of the amplitude spectrum, it is clear that these two sequences also have the same amplitude spectrum.

The figure below shows the amplitude and phase spectra of the minimum phase dipole $(1, 0.5)$ and the maximum phase dipole $(0.5, 1)$. The two amplitude spectra are the same, but their phase spectra are different. We can see from this diagram the origin of the terms minimum and maximum phase.

- For two dipoles with the same amplitude spectra, at any frequency, the magnitude of the phase of the minimum-phase dipole is always less than that of the maximum-phase dipole.

We can extend these ideas from dipoles to time series of length $N$ by decomposing the time series into $N - 1$ dipoles. This leads to the result that there are $2^{N-1}$ possible time series of length $N$ that all have the same amplitude spectra but have potentially different phase
spectra. For only one of these, all the constituent dipoles will be minimum phase. The magnitude of the phase for this series will always be less than the magnitude of the phase for all of the other series. Similarly there will be just one maximum-phase series. All the other series will be mixed phase. The maximum-phase series is simply the minimum-phase series with the elements of all its dipoles reversed in time.

**Correlation as a matrix operation**

Since convolution can be represented as a matrix operation, and correlation involves a correlation with one function time reversed, it is clear that it should be possible to write a correlation as a matrix operation.

If the elements of the cross-correlation of two real time series \{a\} and \{b\} are written as a column vector \( r_{ab} \), then

\[
 r_{ab} = A^T b
\]

where \( A \) is the convolutional matrix containing the elements of \{a\} arranged as in equation (3), and \( b \) is column vector containing the elements of \{b\}.

The auto-correlation of \{a\} is similarly given by

\[
 r_{aa} = A^T a
\]

A *Toeplitz* matrix \( R \), containing the autocorrelation, is also obtained from the operation

\[
 R = A^T A = \begin{bmatrix}
 r_0 & r_1 & r_2 & \ldots \\
 r_1 & r_0 & r_1 & \ldots \\
 r_2 & r_1 & r_0 & \ldots \\
 \vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\]

where \( r_0, r_1, r_2, \ldots \) are the elements of the autocorrelation sequence. Matrices with a Toeplitz structure can be inverted much faster than can general matrices of equivalent size using an algorithm called the *Levinson recursion*. This property is extensively used in seismic processing, especially in Wiener filtering and in predictive deconvolution.

**Minimum and zero phase**

A minimum-phase sequence has the following properties:

- A minimum phase wavelet is causal – it is zero before zero time.
- The minimum phase wavelet has the shortest time duration of all *causal* wavelets with the same amplitude spectra.
- Of all the causal wavelets that have the same amplitude spectrum, the minimum phase wavelet has its energy concentrated as close to the beginning of the wavelet as possible – it has the fastest energy build up.
- The inverse of a minimum-phase wavelet is causal.
- The inverse of a minimum-phase wavelet is stable – it remains finite and tends to zero at long times.
Although the minimum-phase wavelet is the shortest causal wavelet with a given amplitude spectrum, it is not the shortest possible wavelet.

The shortest possible wavelet is the zero phase wavelet; this is the wavelet that has a zero phase spectrum at all frequencies. The zero-phase wavelet is not causal – instead it is symmetric about zero time. It also has the property that its maximum value occurs at that zero time. Although minimum-phase wavelets are important during processing, it is the zero-phase wavelet that is ideal for interpretation of the final stack section and for comparison with wells.
Inverse problems

Many problems in geophysics can be posed as inversion problems. Suppose that we have some set of geophysical observations or data, \( d \). Then, in principle, these data are related to some set of parameters within the Earth that explain them, that is to some Earth model \( m \). Much of geophysics is concerned with trying to recover \( m \) from a given \( d \).

The data \( d \) are related to the model parameters \( m \) by the physics and geometry of the problem, including the characteristics of the sources and receivers of the signals that are being used to make the measurements. In general, we can write this relationship as

\[
A(m) = d
\]  

(11)

where \( m \) and \( d \) are not single variables but are the collection of all the relevant model parameters and observed data, and \( A \) is a function that depends upon the experimental details and fundamental physics, but does not depend upon either the data or the model. Equation (11) therefore represents not just one equation, but a large set of related equations.

This set of equations describes the forward problem. It will be assumed in all cases that we know enough about the physics and the geometry to pose and solve the forward problem computationally. That is, given a model \( m \), we will assume that it is possible to compute theoretical observed data. Typically a forward problem will be represented by a differential or integral equation, or more often by a set of coupled partial differential equations, together with specified initial and/or boundary conditions.

The inverse problem then is to recover \( m \) given \( A \) and \( d \), and we can write it conceptually as a set of equations

\[
m = A^{-1}(d)
\]

In the real world, observations always contain errors – these may be measurement errors, they may be external sources of noise, or they may result from the finite precision with which we can do computer arithmetic. There may also be errors associated with our assumed form for \( A \) – for example, in a seismic problem we may be assuming that the world is acoustic, isotropic, non-dispersive and non-attenuating, whereas the real world may be elastic, anisotropic, dispersive and attenuative. We can regard these imperfections in our forward model also as a type of noise.

Thus, the problem that we really have to solve is to find \( m \) given \( A \) and \( d \) when these are related by the set of equations

\[
d = A(m) + \epsilon
\]

where \( \epsilon \) represents the errors in the observations from all contributions.

Some key elements of real-world inverse problems are:

- All inverse problems are characterised by a large number of parameters that is conceptually infinite – often we will only limit the number of parameters in order to be able to solve the forward problem efficiently.
- The number of observed data is always finite.
- Observed data always contain measurement errors and noise.
• It is impossible to correctly estimate all the model parameters from inaccurate, insufficient and inconsistent data, but it is possible nonetheless to extract useful constraints on these parameters.

• We will almost always have additional prior information about the plausibility of model parameters.

• Most important geophysical problems will be non-linear.

• The computational cost of the forward problem is likely to be significant so that trial-and-error methods are impractical.

• Understanding and quantifying the uncertainty in the model parameters can be as important as estimating their values.

Incomplete, inconsistent and inadequate data

In the solution of any inverse problem, three important questions arise:

1. Does the solution exist?

2. Is the solution unique?

3. Is the solution stable?

From a physical point of view, we know that there must be a solution since we are studying real geological structures in the Earth’s interior. But from the mathematical point of view, given the errors in the observations, there may be no model at all that exactly fits the observed data.

If there is a model that fits the data, say \( A(m_1) = d_1 \), then that solution will not be unique if there is another model \( m_2 \) such that \( A(m_2) = d_1 \). In this case, it will be impossible to distinguish these two models from the given data.

The final question of stability is critical in inversion theory. Suppose that two different models \( m_1 \) and \( m_2 \) generate two different datasets \( d_1 \) and \( d_2 \). Also assume that these two models are very different, while the difference between the two datasets is small and within the noise level \( \epsilon \). In this case, if we attempt to fit the observed data exactly, then the resulting models will vary dramatically but solely as a result of changes in the noise.

In the early part of the twentieth century, it was widely considered that a mathematical problem was formulated correctly if all the three questions posed above had a positive answer. A mathematical problem was considered well-posed if its solution did exist, was unique, and was stable. In contrast, a problem was ill-posed, and was not considered physically or mathematically meaningful, if its solution did not exist, or was not unique, or was not a continuous function of the data (i.e. if for a small perturbation of the data there corresponded an arbitrarily large perturbation of the solution). It turns out that the majority of problems in mathematical physics and geophysics are ill-posed. Fortunately, such ill-posed problems can be both physically and mathematically meaningful, and they can be solved.

Solving matrix equations

In geophysics, we commonly need to solve matrix equations of the form

\[
Ax = b
\]

(12)
where \( A \) and \( b \) are known, and we wish to solve for \( x \). Here, \( x \) is a column vector of length \( M \), \( b \) is a column vector of length \( N \), and \( A \) is an \( N \times M \) matrix with \( N \) rows and \( M \) columns. Equation (12) represents a set of \( N \) linear simultaneous equations in \( M \) unknowns.

If \( M = N \), then \( A \) is a square matrix, and the number equations is equal to the number of unknowns. This *equi-determined* problem has a solution given by

\[
x = A^{-1}b
\]

provided that the matrix \( A \) is not singular – that is provided that the inverse \( A^{-1} \) exists. A singular matrix has a zero determinant and at least one eigenvalue equal to zero. If the matrix is singular, then the simultaneous equations are either not independent or not consistent. For example, the equations

\[
\begin{align*}
x_1 + x_2 &= 1 \\
x_1 + x_2 &= 2
\end{align*}
\]

are not consistent, and the equations

\[
\begin{align*}
x_1 + x_2 &= 1 \\
2x_1 + 2x_2 &= 2
\end{align*}
\]

are not independent. The corresponding matrices

\[
\begin{bmatrix}
1 & 1 \\
1 & 1
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
1 & 1 \\
2 & 2
\end{bmatrix}
\]

are both singular.

In many problems, \( N \neq M \). In this case, \( A \) is a rectangular matrix, and \( A^{-1} \) is undefined.

**Over-determined problems**

If \( N > M \), then we have more equations than unknowns, and if all the equations are independent, then there can be no solution that fits all the equations. However a useful solution can still be found. Such a problem is described as *over-determined*. Now, instead of solving

\[
Ax = b
\]

we solve the related equation

\[
A^T Ax = A^T b
\]

The \( M \times M \) matrix \((A^T A)\) is now square and symmetric, and provided that it is not singular, the solution is

\[
x = (A^T A)^{-1}A^T b
\]

where we can regard the matrix \((A^T A)^{-1}A^T \equiv A^{-N}\) as a type of inverse to the rectangular matrix \( A \) – it is the *least-squares* inverse.

**Under-determined problems**

In other problems, we will have \( N < M \). In this case, we have fewer equations than unknowns. Now the equations do not uniquely define a solution. However in this case too there is still a useful solution to be found. This problem is described as *under-determined*. 


When $N < M$, the matrix $(A^T A)$ will be singular, so we cannot proceed as above. However the $N \times N$ matrix $(A A^T)$ will not be singular if the equations are independent and consistent. We can therefore construct the solution

$$x = A^T (A A^T)^{-1} b$$

where this time the matrix $A^T (A A^T)^{-1} \equiv A^{-M}$ can be regarded as a type of inverse to $A$.

The matrices $A^{-N}$ and $A^{-M}$ are frequently called the *left-hand* and *right-hand* inverses of $A$ since

$$A^{-N} A = I \quad \text{and} \quad A A^{-M} = I$$

but note that

$$A^{-N} A \neq AA^{-M}$$

because their dimensions are different.

### The Euclidean norm

The *Euclidean norm* of a column vector $x$ with $N$ elements, is simply the length of that vector in an $N$-dimensional space. It is written as $\|x\|$. It always has a positive (or zero) value. It is just the generalisation of Pythagoras’ theorem into $N$ dimensions.

In terms of the elements of $x$

$$\|x\| \equiv \sqrt{(x_1^2 + x_2^2 + \ldots + x_N^2)} = \sqrt{\sum_{n=1}^{N} (x_n)^2}$$

The Euclidean norm is frequently used as a measure of the difference between two vectors

$$\|a - b\| = \sqrt{\sum_{n=1}^{N} (a_n - b_n)^2}$$

The square of the Euclidean norm $\|x\|^2$ is the measure that is minimised in least squares techniques. In that context, it is also know as the $L_2$-norm (pronounced “el two”)

$$\|x\|^2 \equiv x_1^2 + x_2^2 + \ldots + x_N^2 = \sum_{n=1}^{N} (x_n)^2 = x^T x$$

Many other norms are possible. The $L_1$-norm is also quite widely used in geophysics

$$\|x\|_{L_1} \equiv |x_1| + |x_2| + \ldots + |x_N| = \sum_{n=1}^{N} |(x_n)|$$

### Linear problems

A linear problem is a problem where changes in the model parameters produce linear changes in the observed data. Such a problem may be described by a set of linear simultaneous equations. Almost all significant geophysical systems are non-linear – that is, changes in the model parameters do not produce linear changes in the observed data. However, most non-linear geophysical problems can be solved by a step-wise sequence of linear approximations. The solution of linear inversion problems is therefore central to almost all geophysical inversion.
A linear problem that generates \( N \) measurements from a model that contains \( M \) parameters can be written as

\[
A \mathbf{m} = \mathbf{d}
\]

where \( \mathbf{d} \) is a column vector containing the \( N \) observed data

\[
\mathbf{d} = (d_1, d_2, d_3, ..., d_N)^T
\]

\( \mathbf{m} \) is a column vector containing the \( M \) model parameters

\[
\mathbf{m} = (m_1, m_2, m_3, ..., m_M)^T
\]

and \( A \) is a matrix with \( N \) rows and \( M \) columns.

In this system, \( \mathbf{d} \) is called the \textit{data vector}, \( \mathbf{m} \) is called the \textit{model vector}, \( A \) is called the \textit{sensitivity} or \textit{condition matrix}, and the linear equations that link them together are called the \textit{equations of condition}.

\textbf{A simple example of linear inversion}

The basic idea of inversion can be captured in a simplified example – suppose that we have to estimate the density structure of the Earth given measurements of its total mass \( M_\odot \) and moment of inertia \( I_\odot \). There are two observational data:

\[
M_\odot = 5.974 \times 10^{24} \text{ kg} \\
I_\odot/a^2 = 1.975 \times 10^{24} \text{ kg}
\]

where \( a \) is the radius of the Earth.

Now we need a model for the density. We will make a number of restrictive assumptions: We will assume that the Earth is spherically symmetric so that density is purely a function of radius. We will assume that the Earth contains two distinct regions – the mantle and the core – within which the densities \( \rho_m \) and \( \rho_c \) are constant. We will assume that the radii of the Earth, \( a = 6371 \text{ km} \), and of the core, \( c = 3485 \text{ km} \), are known exactly. For some models, we will assume that the mantle density and core density are equal, giving a uniform density Earth with \( \rho_m = \rho_c = \rho \).

The table below shows six different possible problems.

<table>
<thead>
<tr>
<th>Data</th>
<th>Model</th>
<th>Data Model</th>
<th>( N )</th>
<th>( M )</th>
<th>Determined</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_\odot )</td>
<td>a uniform Earth</td>
<td>( N = 1 )</td>
<td>( M = 1 )</td>
<td>equi-determined</td>
<td></td>
</tr>
<tr>
<td>( I_\odot/a^2 )</td>
<td>a uniform Earth</td>
<td>( N = 1 )</td>
<td>( M = 1 )</td>
<td>equi-determined</td>
<td></td>
</tr>
<tr>
<td>( M_\odot ) and ( I_\odot/a^2 )</td>
<td>a uniform Earth</td>
<td>( N = 2 )</td>
<td>( M = 1 )</td>
<td>over-determined</td>
<td></td>
</tr>
<tr>
<td>( M_\odot ) and ( I_\odot/a^2 )</td>
<td>a two-layer Earth</td>
<td>( N = 2 )</td>
<td>( M = 2 )</td>
<td>equi-determined</td>
<td></td>
</tr>
<tr>
<td>( M_\odot )</td>
<td>a two-layer Earth</td>
<td>( N = 1 )</td>
<td>( M = 2 )</td>
<td>under-determined</td>
<td></td>
</tr>
<tr>
<td>( I_\odot/a^2 )</td>
<td>a two-layer Earth</td>
<td>( N = 1 )</td>
<td>( M = 2 )</td>
<td>under-determined</td>
<td></td>
</tr>
</tbody>
</table>
In this example, the physics of the forward problem is straightforward. The mass and moment of inertia of a spherically-symmetric earth are given by

\[ M_\odot = 4\pi \int_0^a \rho(r) r^2 dr \]  
\[ I_\odot = \frac{8\pi}{3} \int_0^a \rho(r) r^4 dr \]

(13)  
(14)

The equi-determined case

We will solve problem (4) first. This is an equi-determined problem in which we have a two-layer Earth with two unknown model parameters \( \rho_m \) and \( \rho_c \), and two observations, so \( M = N = 2 \). For a two layer model, the equations relating the model and the observed data are

\[
\begin{align*}
\frac{4\pi c^3}{3} \rho_c + \frac{4\pi}{3} (a^3 - c^3) \rho_m &= M_\odot \\
\frac{8\pi c^5}{15 a^2} \rho_c + \frac{8\pi}{15} (a^3 - c^5) \rho_m &= \frac{I_\odot}{a^2}
\end{align*}
\]

These equations can be written as \( \mathbf{A} \mathbf{m} = \mathbf{d} \) where

\[
\mathbf{d} = \begin{bmatrix} M_\odot \\ I_\odot/a^2 \end{bmatrix}, \quad \mathbf{m} = \begin{bmatrix} \rho_m \\ \rho_c \end{bmatrix}, \quad \text{and} \quad \mathbf{A} = \frac{4\pi}{3} \begin{bmatrix} c^3 & (a^3 - c^3) \\ 2 c^5/5 a^2 & 2/5 (a^3 - c^5/a^2) \end{bmatrix}
\]

Since \( \mathbf{A} \) is a square matrix, then provided that it is not singular, we can solve the problem by finding the inverse of \( \mathbf{A} \), thus

\[ \mathbf{m} = \mathbf{A}^{-1} \mathbf{d} \]

If \( \mathbf{A} \) is singular, then we do not really have an equi-determined system. In the present case, \( \mathbf{A} \) is not singular, and we can invert it to obtain the solution

\[
\begin{align*}
\rho_c &= 12.492 \times 10^3 \text{ kg m}^{-3} \\
\rho_m &= 4.150 \times 10^3 \text{ kg m}^{-3}
\end{align*}
\]

The core is denser than the mantle, as it must be for gravitational stability. There is no incompatibility in the data, and the model predicts the data exactly. There is no redundancy, and we can do no further tests using these data to determine whether this is a realistic model for the Earth.

Now let us look at problem (1), the simplest case where \( N = M = 1 \). Now we have one observation \( M_\odot \) and one unknown model parameter, \( \rho \), the density of a uniform Earth. For constant density, equation (13) gives

\[ M_\odot = \frac{4\pi}{3} \rho a^3 = 5.974 \times 10^{24} \text{ kg} \]

which is easily solved to give

\[ \rho = 5.515 \times 10^3 \text{ kg} \]
Now let us solve problem (2), when again $N = M = 1$. For constant density, equation (14) gives

$$\frac{I_\odot}{a^2} = \frac{8\pi}{15} \rho a^3 = 1.975 \times 10^{24} \text{ kg}$$

which is easily solved to give

$$\rho = 4.558 \times 10^3 \text{ kg}$$

Clearly something must be wrong in this analysis – we have two independent estimates of the density of the Earth that differ by almost 1000 kg m$^{-3}$ which is about a hundred times larger than we would expect from the errors in the observations.

What has happened here is that the model is too simple – in reality, density varies with depth within the Earth. This example illustrates a common problem with many geophysical problems: data error can be less important than errors that arise from uncertainties and inaccuracies in the assumptions behind the model. If our fundamental description of the physics is wrong, or our parameterisation of the model is unable to capture the relevant physics, then the inversion of observed data may appear to give accurate results that are in reality far from correct because the underlying physical model is itself incorrect. If we have no redundant data, then the observations alone cannot reveal such inaccuracies.

**The over-determined case**

In the over-determined problem (3), there are more independent equations than unknowns. In this case, we will not be able to find any model that exactly fits the data. However what we can do is to find a model that minimises the difference between the model predictions and the observed data. Then, for a good model, we would like this difference to be less than the errors in the data.

There are many ways to compare a predicted and an observed dataset. The one that we will use here is to try to minimise the sum of the squares of the differences between the predicted and observed data. That is, if the predicted data are $p$ and the observed data are $d$, then we will try to minimise

$$f = \sum_{j=1}^{N} (d_j - p_j)^2 = \|d - p\|^2 = \|d - Am\|^2 = \sum_{j=1}^{N} \left( d_j - \sum_{k=1}^{M} A_{jk} m_k \right)^2$$

Here $f$ is just a number. It will vary as the model varies, that is $f$ is function of the model vector $m$. $f(m)$ is usually called the *objective function* in geophysics or the *functional* in mathematics.

So now our problem is to find the values for each of the components of $m$ that produce the smallest value for $f$. To do this, we need to differentiate $f$ with respect to each of the model parameters $m_j$, set these differentials equal to zero, and solve the resulting set of $M$ equations for the model parameters $m_j$ where $j = 1$ to $M$. Now we have the same number of equations as unknowns, so that we will get a unique solution – it is the least squares solution to our problem.

When we do this differentiation and reorganise the results as a matrix equation, the equation that we arrive at is

$$A^T Am = A^T d$$

This set of equations is called the *normal equation*. 

36
Now the square $M \times M$ matrix $(A^T A)$ is non-singular, and so the solution that we want is simply

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

Now, returning to problem (3), we have $M = 1$ and $N = 2$, and

$$d = \begin{bmatrix} M_\odot \\ I_\odot/a^2 \end{bmatrix}, \quad m = \begin{bmatrix} \rho \end{bmatrix}, \quad \text{and} \quad A = \begin{bmatrix} 4\pi a^3 \\ \frac{8\pi}{15}a^3 \end{bmatrix}.$$ 

The normal-equation matrix $A^T A$ is now simply a $1 \times 1$ matrix (just a number), so

$$A^T A = \frac{29}{25} \left( \frac{4\pi}{3} a^3 \right)^2$$

The right-hand side of the normal equation is

$$A^T d = \frac{4\pi a^3}{3} \left( M_\odot + \frac{2I_\odot}{5a^2} \right)$$

Since $A^T A$ is actually a scalar, its inverse simply represents division by that scalar, so that

$$\rho = \frac{25}{29} \left( \frac{3}{4\pi a^3} \right) \left( M_\odot + \frac{2I_\odot}{5a^2} \right) = 5.383 \times 10^{-3} \text{ kg m}^{-3}$$

This is the best estimate that we can get for the constant density of the Earth given the two observed data that we have. It is a better estimate than either of the answers to problems (1) and (2) alone.

**Weighting**

So far, we have used $I_\odot/a^2$ as observed data. What happens if we use $I_\odot$ instead? For the equi-determined problem, this will make no difference (provided that we maintain sufficient computational accuracy throughout). However for the least-squares solution there is a difference because now a different quantity will be minimised. Now, we have

$$d = \begin{bmatrix} M_\odot \\ I_\odot \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 4\pi a^3 \\ \frac{8\pi}{15}a^5 \end{bmatrix}.$$ 

Solving these equations as before gives a final answer

$$\rho = 4.558 \times 10^{-3} \text{ kg m}^{-3}$$

This is very different to the answer above; it is exactly the answer that we obtained earlier using only the moment of inertia as observed data. So the answer clearly depends upon how we pose the problem.

What has happened here is that the elements of $A$ are not all in the same units; we have effectively multiplied the second of the normal equations by $a^2$ which is a large number. This equation therefore dominates the least squares solution, and the equation for $M_\odot$ is no longer relevant.
There are two things that we should do to fix this:

- all the observed data should be scaled appropriately so that they are in the same units (or are all dimensionless),
- all the observed data should be divided by their errors (or if we have no information about the errors, then all the data should be scaled to have the same magnitude).

This represents a weighting of the data – if we divide the data by their errors, then we are giving more weight in the solution to the data that are most reliable. If we know nothing about the errors, or if all data are equally (un-)reliable, then we should weight the data to make all the observations the same size.

The under-determined case

A simple example that demonstrates the under-determined problem is that of estimating the spatial distribution of slowness (1/velocity) from measurements of travel times along several ray paths through a solid body. To simplify the problem, we will assume a body containing nine uniform blocks of unit size, arranged in a $3 \times 3$ grid, and we will only consider ray paths that are perpendicular to the block boundaries so that refraction can be ignored and all the ray paths are straight.

\[
\begin{array}{ccc}
T_4 & T_5 & T_6 \\
\hline
x_1 & x_2 & x_3 & T_1 \\
\hline
x_4 & x_5 & x_6 & T_2 \\
\hline
x_7 & x_8 & x_9 & T_3 \\
\end{array}
\]

Now, let the slowness of each block be $x$, and the total travel time across the model be $T$, then the following equations relate the travel times to the slownesses

\[
\begin{align*}
T_1 &= x_1 + x_2 + x_3 \\
T_2 &= x_4 + x_5 + x_6 \\
T_3 &= x_7 + x_8 + x_9 \\
T_4 &= x_1 + x_4 + x_7 \\
T_5 &= x_2 + x_5 + x_8 \\
T_6 &= x_3 + x_6 + x_9
\end{align*}
\]

Given the six measurements of $T_1$, $T_2$ ... $T_6$, the inverse problem is to determine information about the nine slownesses $x_1$, $x_2$, ... $x_9$. 

38
The equation of condition for this system is

\[
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7 \\
x_8 \\
x_9 
\end{bmatrix}
= 
\begin{bmatrix}
T_1 \\
T_2 \\
T_3 \\
T_4 \\
T_5 \\
T_6 
\end{bmatrix}
\]

We can see here why the matrix \( A \) is called the sensitivity matrix. The element in the \( i \)th row and \( j \)th column gives the sensitivity \( (\partial T_i / \partial x_j) \) of the \( i \)th measurement to a change in the \( j \)th model variable. So, for example, the fourth measurement is only sensitive to \( x_1 \), \( x_4 \) and \( x_7 \), and it is equally sensitive to each of these variables. Thus \( (\partial T_4 / \partial x_j) = 1 \) for \( j = 1, 4, 7 \), and = 0 otherwise. Note that when \( (\partial T_i / \partial x_j) = 0 \), a change in the slowness \( x_j \) will not affect the value of the travel time \( T_i \), thus we can find no information about the value of \( x_j \) from the measured value of \( T_i \).

If we want to visualise the sensitivity for a particular measurement, then it is convenient to display the corresponding row of the sensitivity matrix in a natural ordering, that is in an order that corresponds to the spatial distribution of the parameters. For the second travel time \( T_2 \), this display would be

\[
\begin{bmatrix}
0 & 0 & 0 \\
1 & 1 & 1 \\
0 & 0 & 0
\end{bmatrix}
\]

and for the fourth travel time \( T_4 \), the display would be

\[
\begin{bmatrix}
1 & 0 & 0 \\
1 & 0 & 0 \\
1 & 0 & 0
\end{bmatrix}
\]

We can also display the model parameters using the same geometry. For large models of course, we would not display the values as numbers, but instead display the same information graphically using a suitable colour scale.

Now suppose that there are no errors in the observations, and that the travel time in each of the columns and rows is equal to 6 units (i.e. \( T_i = 6 \)). Then clearly a homogeneous model, for which the slowness in each block is 2, will satisfy the data exactly. However the following models also satisfy the data exactly

\[
\begin{bmatrix}
1 & 2 & 3 & -2 & 0 & 8 & 2 + \alpha & 2 & 2 - \alpha & 2 + \beta & 2 - \beta & 2 \\
2 & 2 & 2 & -2 & 6 & 2 & 2 & 2 & 2 & 2 - \beta & 2 + \beta & 2 \\
3 & 2 & 1 & 10 & 0 & -4 & 2 - \alpha & 2 & 2 + \alpha & 2 & 2 & 2
\end{bmatrix}
\]

where \( \alpha \) and \( \beta \) are arbitrary constants.
In this problem therefore, there are infinitely many models that satisfy the data. Some of these, such as the second, may not satisfy other constraints on the model parameters – for example, the slowness cannot be negative. But even with external constraints, in a problem such as this there will still be infinitely many models that satisfy the data exactly.

The null space

The null space of the sensitivity matrix $A$ is defined as the set of all column vectors $m$ such that $Am = 0$. It is easy to show that each of the following models represents vectors in the null space of $A$ for this problem

$$
\begin{bmatrix}
0 & 1 & -1 \\
0 & -1 & 1 \\
0 & 0 & 0 \\
1 & -1 & 0 \\
-1 & 1 & 0 \\
0 & 0 & 0 \\
0 & 1 & -1 \\
-1 & 1 & 0
\end{bmatrix}
$$

In fact, these four models represent a basis for the null space of $A$. That is, any model in the null space of $A$ can be written as a linear combination of these four models.

The significance of the null space is that, if $v$ is any vector in the null space, $\alpha$ is any number, and $m$ is any model that satisfies $Am = d$, then the model $m + \alpha v$ also satisfies the observed data $d$. In other words, we can add any linear combination of vectors from the null space to a model that satisfies the data, and still satisfy the data.

Inexact incomplete data

As all measurements are noisy, let us assume that the following measurements were made for the travel times

$$(6.07, 6.07, 5.77, 5.93, 5.93, 6.03)$$

Now, even though there are fewer data than model parameters, there is no model that fits the data exactly.

We can see this because

$$T_1 + T_2 + T_3 \quad \text{should be equal to} \quad T_4 + T_5 + T_6$$

but from the data, we have

$$T_1 + T_2 + T_3 = 17.91 \quad \text{and} \quad T_4 + T_5 + T_6 = 17.89$$

so that, with these data, there can be no solution.

How then should we proceed? The solution is to use a least-squares approach for underdetermined systems, and solve using

$$x = A^T (AA^T)^{-1} d \quad (16)$$

Using this approach, we can find a solution that has the minimum data mismatch. Such a solution is

$$\begin{bmatrix}
2.011 & 2.011 & 2.044 \\
2.011 & 2.011 & 2.044 \\
1.911 & 1.911 & 1.944
\end{bmatrix}$$
This however is not the only best-fitting least-squares solution. There is still a null space, and an infinite number of solutions will satisfy the data equally well.

The solution that we have found using (16) minimises the least square error, and having satisfied that constraint then minimises the $L_2$ norm of the model $\| \mathbf{m} \|^2$. The latter condition makes the model as “short” as possible given that it must also match the data. This is a minimum model that has nothing within it that we can leave out without degrading the fit to the data. However we can choose to add to it any linear combination from the null space.

In many circumstances, it is appropriate to parameterise the model so that the parameters that we are attempting to obtain are not defined in an absolute sense, but are defined as changes to some initial model. If we parameterise in this way, then a solution using (16) will ensure that we find the model that best fits the data and that is also as close to the starting model as possible. In this case, the problem is often set up to solve

$$\mathbf{m} - \mathbf{m}_0 = A^T (A A^T)^{-1} (\mathbf{d} - A \mathbf{m}_0)$$

where $\mathbf{m}_0$ is the starting model, and $(\mathbf{d} - A \mathbf{m}_0)$ is the difference between the observed data and that predicted using the starting model.

**Least squares filters**

Wiener filtering is a form of least-squares inversion applied to one-dimensional time series. Apart from very simple operations such as scaling, muting, shifting, sorting, stacking and perhaps band-pass filtering, the Wiener filter is probably the most widely used of all operations in seismic processing.

**Least-squares inversion of a minimum-phase dipole**

We have already seen that one of the problems of inverting a dipole via an expansion of the denominator as a series is that the inverse filter can result in a long operator. This is especially true when there is a zero close to the unit circle.

The problem that we would like to solve here is to find a filter of useable finite length, which when convolved with a dipole, produces an output that resembles as closely as possible a single unit spike at zero time. We will invert a minimum-phase dipole $\{a\} = (1, \alpha), \; |\alpha| < 1$, and we will assume that $\alpha$ is real.

For the ideal inverse filter, we have

$$A(z) F(z) = 1$$

For a filter of finite length, we will have

$$A(z) F(z) \approx 1$$

In the time domain, we will have

$$(1, \alpha) \ast (f_0, f_1, f_2, \ldots) = (x_0, x_1, x_2, x_3, \ldots) \approx (1, 0, 0, 0)$$

which can be written in matrix form as

$$\begin{bmatrix}
1 & 0 & 0 & \cdots \\
\alpha & 1 & 0 & \cdots \\
0 & \alpha & 1 & \cdots \\
0 & 0 & \alpha & \cdots \\
\vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\begin{bmatrix}
f_0 \\
f_1 \\
f_2 \\
\vdots
\end{bmatrix}
\approx
\begin{bmatrix}
x_0 \\
x_1 \\
x_2 \\
\vdots
\end{bmatrix}
\begin{bmatrix}
1 \\
0 \\
0 \\
\vdots
\end{bmatrix}$$
or as

\[ \mathbf{A} \mathbf{f} = \mathbf{x} \approx \mathbf{b} \]

where \( \mathbf{A} \) is a matrix that contains the dipole positioned and padded to represent a convolution, \( \mathbf{f} \) is the unknown column vector that contains the elements of the inverse filter that we are trying to find, and \( \mathbf{x} \) is the actual output and \( \mathbf{b} \) is the ideal output of the filter.

Now we would like to find the elements of the filter \( \mathbf{f} \) that minimises the error. That is, we want to minimise

\[
\sum_n (b_n - x_n)^2 = \sum_n \left( b_n - \sum_m f_m a_{n-m} \right)^2 = \| \mathbf{b} - \mathbf{A} \mathbf{f} \|^2
\]

This is an over-determined least-squares problem, and its solution is obtained by solving

\[ \mathbf{A}^T \mathbf{A} \mathbf{f} = \mathbf{A}^T \mathbf{b} \quad (17) \]

In this case however, the term \( \mathbf{A}^T \mathbf{A} \) in equation (17) is just the auto-correlation matrix \( \mathbf{R} \) of \( \{a\} \), the next term is the unknown filter coefficients that we want to find, and the term on the right-hand side is the cross-correlation of \( \{a\} \) with the ideal output \( \{b\} \). The first and third of these can be calculated easily, so that we have \( M \) linear equations in the \( M \) unknown variables \( f_m \). The solution is simply

\[
\mathbf{f} = \mathbf{R}^{-1} \mathbf{A}^T \mathbf{b}
\]

where all the terms are straightforward to calculate. Thus this approach, which is called \textit{Wiener filtering}, can invert a minimum phase dipole to give a filter with a finite number of coefficients that does the best job in a least squares sense.

The story does not end here – the matrix \( \mathbf{R} \) has a \textit{Toeplitz} structure

\[
\mathbf{R} = \begin{bmatrix}
1 + \alpha^2 & \alpha & 0 & \cdots \\
\alpha & 1 + \alpha^2 & \alpha & \cdots \\
0 & \alpha & 1 + \alpha^2 & \cdots \\
\vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\]

Among other properties, it is clear that only one row of the matrix needs to be stored to define all its elements. More importantly, a fast algorithm, the \textit{Levinson recursion} is available to invert such a matrix. Normally the inversion of an \( M \times M \) matrix requires of the order \( M^3 \) operations; for a Toeplitz matrix, only of the order of \( M^2 \) operations are required.

\textbf{Wiener filters}

The approach above can be extended to find the least-squares inverse filter for any minimum-phase sequence. It can also be extended to find a filter that is designed to produce a desired output \( \mathbf{b} \) that is not simply a spike at zero time followed by zeros, but is any appropriate time series. The design of such a filter, and its application by convolution to seismic or similar data, is called \textit{Wiener filtering}.

In all cases, design of the filter involves solving the equation

\[
\mathbf{f} = \mathbf{R}^{-1} \mathbf{A}^T \mathbf{b}
\]

where \( \mathbf{R} \) is a Toeplitz matrix containing the auto-correlation of the signal that we want to invert, and \( \mathbf{A}^T \mathbf{b} \) is its cross-correlation with the desired output of the filter.
Wiener filtering has many applications, but three of these are especially useful:

**Spiking deconvolution**

The principal purpose of spiking deconvolution is to replace a known source wavelet within a seismic trace by a simple spike. Thus, if we assume that the seismic trace is composed of a one-dimensional reflectivity sequence convolved with a source wavelet, then application of a spiking filter will deconvolve out the source wavelet leaving only the reflectivity sequence.

In its simplest application, we need to know the auto-correlation of the source wavelet, and the cross correlation of the source wavelet with a spike. If we know the source wavelet, either because we have measured it in the field or we have been able to calculate it from physics and engineering principles, then we can find the filter and apply it to the seismic data. Such an approach is one form of deterministic deconvolution.

However, in many cases we will not know the source. In these cases, provided that the source is minimum phase, we can recover it from the data using some plausible assumptions. First we assume that the noise in the data is uncorrelated. Consequently its auto-correlation is just a spike at zero lag. Second, we assume that the reflectivity sequence (i.e. the geology), is uncorrelated. Therefore its auto-correlation will also be a spike at zero lag. And if the observed seismic data is simply the convolution of the geology with a source wavelet plus noise, then the auto-correlation of the seismic data will be identical to the auto-correlation of the source. This in turn is just the Fourier transform of the square of the amplitude spectrum, so if we knew the phase spectrum of the source, then we would know the source wavelet.

The minimum-phase assumption is sufficient to close this loop – it uniquely defines the phase spectrum given the amplitude spectrum. Thus, provided that the source is minimum phase, we can extract it from the data, design the appropriate Wiener filter to replace it by a spike, and deconvolve it from the data.

Despite the apparently restrictive nature of the assumptions that go into this method, in practice it works surprisingly well for the vast majority of seismic data. The principal practical problem relates to limitations in the bandwidth of the seismic data and to notches in the spectrum of the source wavelet. Their solution involves stabilising the design of the Wiener filter, as discussed below.

**Shaping filters**

A spike has a very broad frequency bandwidth – the broadest that is possible for a given sample interval. Spiking deconvolution is therefore attempting to flatten, broaden and whiten the amplitude spectrum of the output completely.

Real seismic data has limited bandwidth, and often has notches in the spectrum associated with destructive interference, for example produced by ghost reflections from the sea surface. In such cases, spiking deconvolution, will cause the noise in these low-amplitude parts of the spectrum to blow up to a degree that they may dominate the resulting data.

A more realistic approach is to recognise that the bandwidth is limited, and then to attempt to replace the real wavelet, not by a spike but by some other wavelet that is more desirable than the real source wavelet, but that has a similar bandwidth to the data – perhaps a zero-phase wavelet. Such filters are often called shaping filters – they attempt to shape the source wavelet into some desirable form.
The design of a Wiener shaping filter is straightforward. In this case, the desired output vector \( \mathbf{b} \) is not a spike but is the finite-duration desired wavelet. The desired wavelet need not be minimum phase, and it need not be causal, and the filter itself need not be causal. There will however still be a minimum phase assumption associated with the extraction of the original source from the data if it is not known \( a \) priori.

**Predictive deconvolution**

One of the most widely used versions of the Wiener filter is in *predictive* or *gapped* deconvolution. In this application, we first design a Wiener filter \( \mathbf{f} \), of length \( N \), where the desired output of the filter is the input data at \( n \) time steps into the future. The time \( n \Delta t \) is called the *prediction distance* or the *prediction lag*.

To design such a filter, the auto-correlation used is just the auto-correlation of the input seismic data, and the cross-correlation used is that between the seismic data and the same data shifted in time by the prediction lag. If such a filter were then applied to the input data, the output would be only those parts of the data that were predictable using the input from \(-n\) to \(-(n + N)\) samples earlier in the time series.

Now, we actually apply to the data not this filter, but the related filter

\[
(1, 0, 0, ..., 0, -\mathbf{f})
\]

That is, we apply a filter with a unit spike followed by \( n - 1 \) zeros followed by the negative of all the coefficients of the filter \( \mathbf{f} \).

What does such a filter do? It will try to remove anything from the data that is predictable with a lag of between \( n \) and \( n + N \) time steps. Source reverberations and multiple reflections should be highly predictable; the filter will therefore attempt to remove them from the input data. Predictive deconvolution is used for two principal purposes: to remove short-period, reverberatory multiples, typically generated at the seabed in shallow water, and to shorten or *compress* the source wavelet. Both these operations, and especially the latter, will whiten the frequency spectrum and so will tend to introduce noise at missing frequencies.

Typically the lag is chosen to be larger than the desired source wavelet length, and less than the period of the shortest multiple. Choice of the filter length affects the maximum multiple period that can be suppressed. In general, least-squares filter design degrades as the filter gets longer, and there are therefore maximum practical multiple periods that can be suppressed by this method. For most datasets the maximum effective period is a few 100 ms. Below this period, predictive deconvolution, though simple in concept, can be remarkably stable and effective. It can be especially effective if the input seismic data are first manipulated so that they better match the assumptions of the method – minimum-phase source wavelet and multiple generation, random “white” geology, random stationary noise, and a fixed relationship between signal and its associated multiples and reverberations.

**Stabilisation, damping, regularisation**

Thus far, we have assumed, when designing Wiener filters or inverting geophysical data, that there will be no issues (apart from singularity) associated with inverting square matrices. There are of course issues.

A matrix may not be singular, but it may still have a very small determinant and at least one very small eigenvalue. In such a case, although the inverse may exist, the problem is unstable – large spurious values will likely appear in the solution. The solution will depend
upon the values of the smallest eigenvalues, which themselves will likely be almost entirely related to the noise or errors in the system. Effectively, in any inversion problem, we face the issue of potentially dividing by small numbers that are noise dominated.

We have already seen this problem when considering deconvolution in the frequency domain. In that case, we stabilised the system by adding a small positive value to the denominator. In matrix inversion, we do something that is closely analogous. Now, instead of minimising the objective function

\[ f = \|d - Am\|^2 \]

we instead try to minimise the objective function

\[ f_\mu = \|d - Am\|^2 + \mu \|m\|^2 \]

where \( \mu \) is a positive scalar known in various contexts as the damping factor, trade-off, regularisation or pre-whitening parameter.

When we now invert, we find that the equations we must solve are

\[ m = (A^T A + \mu I)^{-1} A^T d \]

For a positive \( \mu \), the matrix \((A^T A + \mu I)\) is always non-singular; it can be inverted to solve equi-, under-, over- and mixed-determined systems even if \( A^T A \) and \( A A^T \) are singular.

By damping the least squares system, we have protected the solution against small eigenvalues by adding a constant value, controlled by \( \mu \), to the diagonal of \((A^T A)\). If \( \mu \) is very small, then we will have regular least squares; if \( \mu \) is very large, then we will simply minimise the norm of the model parameters. Choosing the value of \( \mu \) involves a trade off between fitting the data and stabilising the result.

**Non-linear and linearised inversion**

Most real inversion problems in geophysics are non-linear. A few such problems (not many) have special tricks that can be used to solve them. A few have sufficiently simple forward problems, or sufficiently restricted possible parameter values, that they may be realistically solved by some form of intelligent trial-and-error method – for example using a simple Monte Carlo approach or a more sophisticated directed search using a method such as genetic algorithms in which a suite of models evolves in time in a way that has biological analogues. These methods attempt to find the global solution.

In all other cases, the approach taken is to use linearised inversion in which a series of small optimised linear steps are taken in order to solve the underlying non-linear problem.

In a non-linear problem, we are trying to solve

\[ F(m) = d \] (18)

where \( m \) and \( d \) can be regarded as vectors, but \( F \) is not a matrix – rather it is whatever computational or mathematical process we have to follow in order to generate synthetic data from a model. For example, in seismics, it might involve simulating the propagation of waves through a heterogeneous model.

Linearised inversion involves iterating the linear problem

\[ Am = d \] (19)
where the elements of the matrix $A$ are $(\partial p_i / \partial m_j)$ where $p$ is the predicted data.

Linearised inversion proceeds from a starting model. This model is used in equation (18) to generate synthetic data. These synthetic data are used to construct the partial derivatives that are required to form the matrix $A$, and equation (19) is solved using the methods of linear inversion to obtain a new model. This new model is used to predict new synthetic data, and the process is iterated until sufficiently close agreement is reached between the predicted and observed data.

There can be a very large computational cost associated with calculating the partial derivatives required in equation (19), and a number of general methods exist that are applicable to systems that involve propagating or diffusing signals that allow the calculation of these derivatives efficiently. Typically these involve the forward propagation of signals from the sources into the model, and the back propagation of the data errors from the receivers. These two fields are then combined at every point in the model (typically they are cross-correlated in some way) to generate the required partial differentials. Such methods are called adjoint methods. All forms of tomography use this approach including seismic travel-time, x-ray, electro-magnetic, resistivity, radar-attenuation, and seismic-wavefield tomography. Reverse-time migration also falls into this category.

Linearised inversion as described here is a simple gradient method. It seeks to find a minimum on a multi-dimensional error surface by continually heading downhill (in many dimensions) until the bottom of the valley is found. More sophisticated methods exist that seek in various ways to learn about the shape of this surface in order to optimise the route taken to the minimum, for example conjugate gradients.

All these methods though are local methods – they seek to find the bottom of the nearest local minimum in the error surface which will not necessarily be the global minimum. Consequently they typically require good starting models that are in the general vicinity of the global minimum. They are also sensitive to the characteristics of the error surface – broad simple, single minima are easy to find; narrow minima on a flat plane, minima surrounded by many other shallower near-by minima, and broad, flat minima with small irregularities will be very difficult to discover. For linear problems, the error surface is a simple parabola.

Except where special algorithms are available, for example the Levinson recursion for Toeplitz matrices, matrix equations are not normally solved by formal inversion of a matrix since this is an expensive operation for large matrices. Instead there are a number of both direct and iterative methods that are able to solve matrix equations without formally inverting the relevant matrix. These methods are especially relevant when the matrix to be inverted is sparse – that is, when many of its elements are zero. This will almost always be the case in large physical problems.