Geophysical Journal International

Geophys. J. Int. (2010) **182,** 1001–1012 doi: 10.1111/j.1365-246X.2010.04658.x

Approaches to automated data selection for global seismic tomography

Andrew P. Valentine and John H. Woodhouse

Department of Earth Sciences, University of Oxford, OX1 3PR, *UK. E-mail: andrew.valentine@earth.ox.ac.uk*

Accepted 2010 May 11. Received 2010 May 10; in original form 2010 March 20

SUMMARY

The ability to handle large amounts of data automatically is essential for any major tomographic inversion. As part of this process, it is necessary to differentiate between high-quality seismograms, and those that are unusable due to noise or other errors. This quality assessment is traditionally made visually; however, the sheer quantity of data in a modern tomographic data set makes this approach unfeasible. It is therefore necessary to develop techniques for automating this quality assessment process.

We demonstrate that a simple neural network, trained to recognize the frequency-domain characteristics of high- and low-quality data, can successfully distinguish the two classes in unseen data. We demonstrate that the resulting clean data sets are of sufficient quality to allow full-waveform determination of event focal mechanisms and hypocentral parameters.

The process we outline allows the rapid creation of a high-quality data set for seismic tomography. Depending on application, this may be suitable for use without further refinement. In some circumstances, a further visual inspection may remain desirable to ensure the data set is noise-free; however, a significant benefit will still derive from the reduction in number of traces to be examined. This will enable full-waveform inversion using significantly larger data sets than has hitherto been possible. The selection strategy relies only on measurements made from the seismogram, and on rough estimates of hypocentral location—the final data set does not depend on any *a priori* assumptions regarding earth structure or wave propagation.

Our focus has been on data selection for seismic tomography, but the approach is general and may find application across a wide range of seismic investigations. An automated system is of interest wherever large data sets must be handled, or where time is of the essence—such as in earthquake hazard assessment.

Key words: Neural networks, fuzzy logic; Earthquake source observations; Seismic tomography.

1 INTRODUCTION

The creation of robust and credible data sets is one of the most important and time-consuming tasks for an observational seismologist. Large quantities of seismic data are now routinely available from national and international data centres, but the quality of this data varies enormously—background noise, instrument malfunction, or errors introduced during data storage and retrieval may all render a given seismogram unusable. As a result, blind use of all available data relevant to a particular event or region is impossible: for many applications the poor-quality data will distort results significantly. Consequently, the seismologist must develop strategies to accommodate this, either removing 'bad' traces from the data set before use, or rejecting them during processing.

For many, the gold-standard is a visual inspection of each trace to be used. However, this is extremely time-consuming, and severely limits the amount of data that can be used. It is straightforward to obtain over 2500 seismograms for a recent event; even restricting ourselves to using only seismic stations in major global networks may result in several hundred traces. Visual inspection of any significant number of events is therefore difficult in practice. In addition, the inherently subjective nature of such processing leads to doubts over uniformity and repeatability. Automating the data selection process would circumvent these problems, and allow the rapid generation of extremely large data sets, with limited need for human intervention.

Obviously, any assessment of data quality must be informed by an understanding of the intended uses for the data: the exact definition of an 'acceptable' seismogram will vary. As a result, any attempt to design an automatic data selection strategy must be linked to a particular goal, although the resulting system may be applicable in other situations; our interest lies in generating data sets suitable

for full-waveform tomography. The data used in these inversions consists of waveforms extracted from seismograms and filtered in particular frequency bands. Extraction is performed automatically, using simple traveltime calculations. The main data selection task is then to assess whether each trace should be retained in the data set, or discarded.

Two basic approaches to this problem are available to us. First, we could make use of forward modelling to assess how well each waveform extracted from a real seismogram corresponds to our expectations. This appears a straightforward solution, and may perform tolerably well in many cases. However, it leads to the data set having a dependence on whatever method and parameters have been used for forward-modelling—which may include particular earth models, and information about seismic sources that have themselves been derived from analysis of some unknown data set. This is undesirable, particularly in the current case, where we intend to generate a new earth model from the data. The alternative is to define some measure of the *a priori* quality of a waveform, and use this to differentiate between acceptable and unacceptable traces. This attempts to mimic the seismologist's approach when performing visual classification, and can be free from any external assumptions.

A range of techniques have been proposed for automatic handling of seismic data, combining elements of both these approaches. A common application is the automatic determination of moment tensors for earthquakes, and a number of schemes exist to accomplish this. Many of these involve methods for stable computation of focal mechanisms, taking into account that some proportion of the data is likely to be unusable, and do not attempt to produce a smaller, high-quality data set. At the simplest, this may entail using data only from stations with a history of providing high-quality data (Pasyanos *et al.* 1996); alternatively, an adaptive weighting scheme may be used that progressively down-weights data that do not agree with the recovered mechanism (e.g. Bernardi *et al.* 2004). A similar approach is taken in other automated seismological applications, such as event location (Bolt 1960). Where data quality is considered prior to moment tensor determination, this is generally done via analysis of signal-to-noise ratios (e.g. Scognamiglio *et al.* 2009).

Another, related, application involves automatic detection of particular phase arrivals, perhaps as part of an automatic windowing system: inherently, this process involves some degree of quality assessment. Again, a variety of methods exist, ranging from the detection of sudden increases in activity in a seismogram (Allen 1978; Baer & Kradolfer 1987) to neural networks trained to distinguish desired arrivals from background noise (Dai & MacBeth 1995). Enhancements such as error estimation and assessment of quality via signal-to-noise ratios may also be incorporated (Di Stefano *et al.* 2006). Of particular relevance to our current work is the FLEXWIN system developed by Maggi *et al.* (2009), intended to assist in the selection of measurement windows based on evidence of phase arrivals, and comparisons between data and synthetic seismograms. These windows may then be used to perform tomographic inversions, as demonstrated by Tape *et al.* (2009). Indeed, the need to handle large data sets is common to other tomographic studies, and some have incorporated a high degree of automation. This usually has relied on an assessment of whether measurements made on individual seismograms are sufficiently robust to be usable (e.g. van Heijst & Woodhouse 1997, 1999; Lebedev *et al.* 2005; Lebedev & van der Hilst 2008). For the case of full-waveform studies, this would entail reliance on forward-modelling, with the inherent problems discussed earlier.

These systems are all designed to enable processing of data sets containing unusable data, rather than explicitly attempting to remove poor traces. Of course, the two tasks are linked, and it is possible to adapt such systems to produce a reduced data set. However, it would be preferable to have a system designed explicitly for quality assessment. Ideally, this should be performed in a manner that requires little or no information beyond that contained in individual seismograms. Since the characteristics of the seismogram depend strongly on the depth of the seismic event, and on the distance between source and receiver, some additional information about these parameters may prove useful during classification. Obviously, seismometer locations are easily obtained, and can be assumed to be correct; estimates of source location from, for example, traveltime measurements are also available. This need not introduce any unacceptable bias into our quality assessment: the sensitivity of such measurements to plausible variations in earth model is far below the level at which there would be an appreciable impact on our determination.

Our experience dictates that the focus should be on removing poor-quality traces, rather than attempting to identify a high-quality subset of the data. In many cases, the characteristics that make a particular waveform unusable are more easily defined than those making another waveform satisfactory; in addition, such an approach fosters realistic expectations of the end result. It should be assumed that even the 'clean' data set may contain traces that a seismologist would reject on visual inspection, and this must be accommodated when the data set is used.

We begin by outlining the data to be used in this study. Using examples of visually good and bad waveforms we show that the spectral characteristics of a particular waveform provide a good indicator of data quality, and that it is possible to use a neural network approach to classification. We therefore outline the basic operation of a 'mulitlayer perceptron', and demonstrate that this can be applied to the current problem. Finally, we illustrate the effectiveness of this approach by implementing fully automatic moment tensor determination.

2 SEISMIC WAVEFORMS: GOOD AND BAD

For the purposes of this paper, we will consider two classes of waveform; extension to further classes is straightforward. The classes we describe here are commonly used in full-waveform studies (e.g Dziewonski *et al.* 1981; Woodhouse & Dziewonski 1984; Thurber & Ritsema 2007). Each class is defined in terms of:

(i) a time-window rule, specifying the start and end times of the waveform relative to the event time and
(ii) a set of filtering operations.

Together, these allow the extraction of the waveforms from raw seismic data. The rules may be complex: we require only that waveforms are generated in a consistent fashion, so that any relevant patterns may be observed.

The first class of data is the long period body wave ($f \lesssim \frac{1}{45}\ \mathrm{s}^{-1}$) part of the traces, as used in CMT analysis (Dziewonski *et al.* 1981; Dziewonski & Woodhouse 1983). We define this to correspond to the portion of the seismogram from the event time to 5 min before the theoretical surface wave arrival time, assuming great-circle propagation and a surface wave speed of 4 km s^{-1}. We convert the resulting time-series to an SRO instrument response (Fig. 1), and apply a cosine bandpass filter with corner frequencies at 1, 2, 16.7 and 18.5 mHz. Examples of plausible and noisy waveforms from this class from a range of events are shown in Fig. 2.

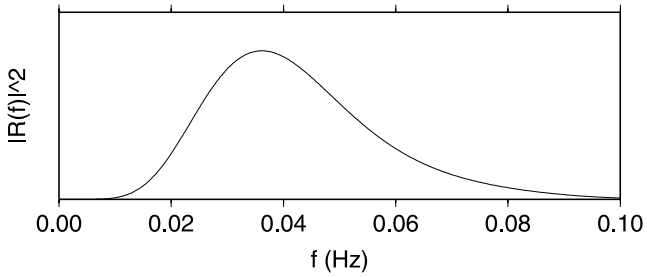

Figure 1. SRO instrument response function.

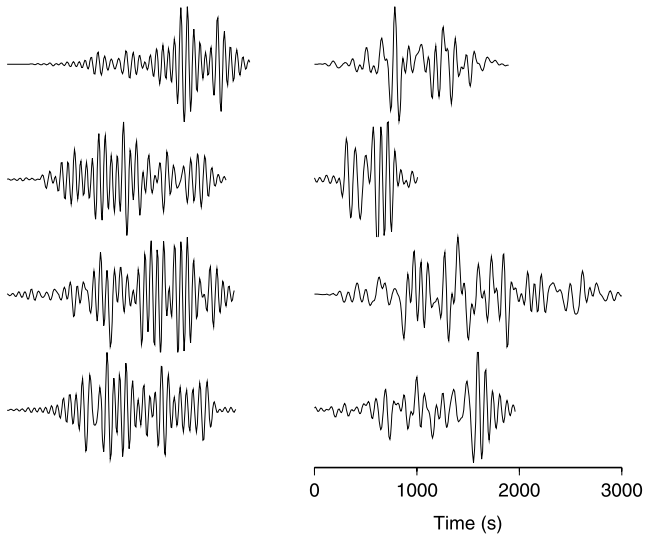

Figure 2. Examples of waveforms in the body-wave class. Left-hand column shows plausible seismograms; right-hand column contains traces identified visually as being unusable. The horizontal scale is same for all traces; vertically, all traces have been scaled to have unit maximum amplitude.

Our second class is similar, but is intended to be sensitive to mantle waves ($f \lesssim \frac{1}{135}$ s^{-1}). The time window is chosen to extend from the event time, until 30 min after the theoretical arrival time for the third surface wave orbit. The time-series is again converted to an SRO response, and bandpass filtered with corners at 1, 2, 6.7 and 7.4 mHz. Some example waveforms can be seen in Fig. 3.

As implied above, a purely visual classification involves identifying 'plausible', rather than 'correct' traces. Without further information, we cannot necessarily recognize all possible sources of error—for example, modest timing errors may not be obvious until synthetic traces are available for comparison. We also note that the distinction between plausible and noisy seismograms is not always clear-cut, even for the experienced seismologist; quality assessment is made on a continuous scale. This must be borne in mind when considering the effectiveness of any automatic algorithm: 'misclassifications' may actually represent inconsistencies in the visual assessment.

Clearly, the key to building an automatic data classification system lies in identifying characteristics that may be used to differentiate plausible data from noise; for an automatic system, these need to be expressed as measurable quantities. An advantage of taking a neural network approach to classification is that we do not need to fully understand the relationship between these parameters and the desired classification—this can be 'learned' by the network—but we must decide which information is provided to the network. Since the amount of 'training' required by the network will grow

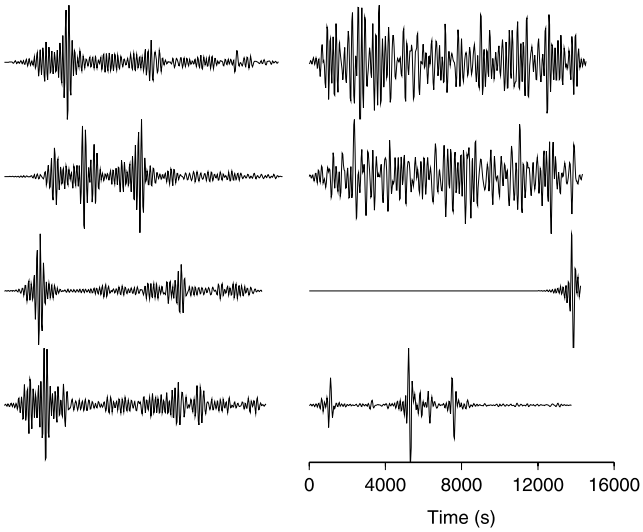

Figure 3. As Fig. 2, but for the mantle-wave class. Again, left-hand column contains plausible traces; right-hand column traces are unusable.

with the number of input parameters, we wish to keep this as small as possible.

One approach would be to use the time domain representation of the waveform—this is the information used when a visual classification is made, and must therefore be sufficient to determine whether a waveform is plausible. However, two factors suggest that this is not a fruitful avenue to pursue. First, the number of parameters required to accurately represent several hours of waveform in the time domain is large—around 10^3. This alone is likely to be prohibitive; however, we also know that the time-domain form of a seismogram is an extremely complex function of source and receiver locations, event mechanism, and earth model. It therefore seems unlikely that a straightforward, universal measure of waveform quality can be derived from this.

As an alternative, we can use the frequency domain representation of the waveforms; as is well-known, this need not entail any loss of information. Furthermore, by working in the frequency domain, we restrict ourselves to a frequency band specified in the class definition; in the time domain, we must be prepared to handle time-series of varying length, making implementation significantly more complex.

Figs 4 and 5 show the power spectra corresponding to the various exemplar seismograms shown in Figs 2 and 3. We see that the noisy waveforms contain significantly more power at low frequencies than the plausible ones, although the overall shape of the power spectrum is closely related to that of the SRO filter. To illustrate this further, Fig. 6 shows the average power spectra of a set of waveforms drawn from a number of events, and classified by hand (the 'training data set', as described in Section 4.2). The difference between the spectra of good and poor data is clearly seen. Our quality assessment problem therefore becomes one of determining which category best describes a given spectrum. This type of pattern-matching problem is typically well-handled by a neural network (e.g. Bishop 1995), and we explore such an approach.

3 NEURAL NETWORKS

Neural networks are increasingly used for a wide variety of classification and data processing applications. As the name implies, they may be seen as a model for how the brain processes information, and

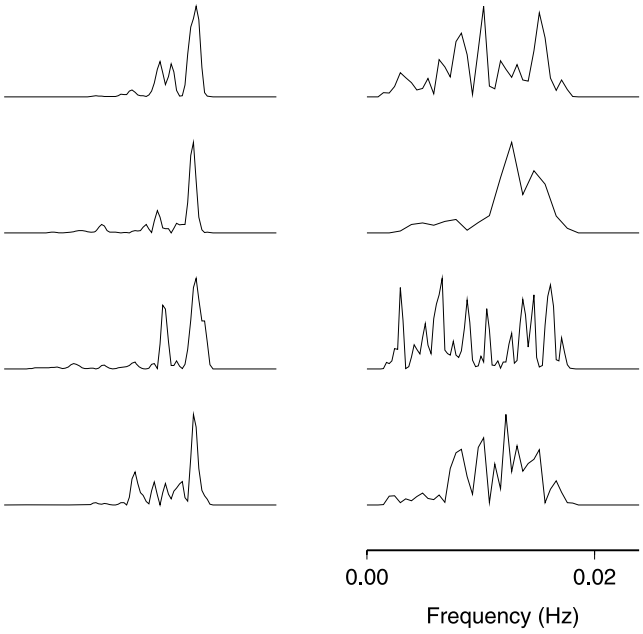

Figure 4. Power spectra for the eight waveforms shown in Fig. 2. All spectra have been normalized to have unit maximum amplitude.

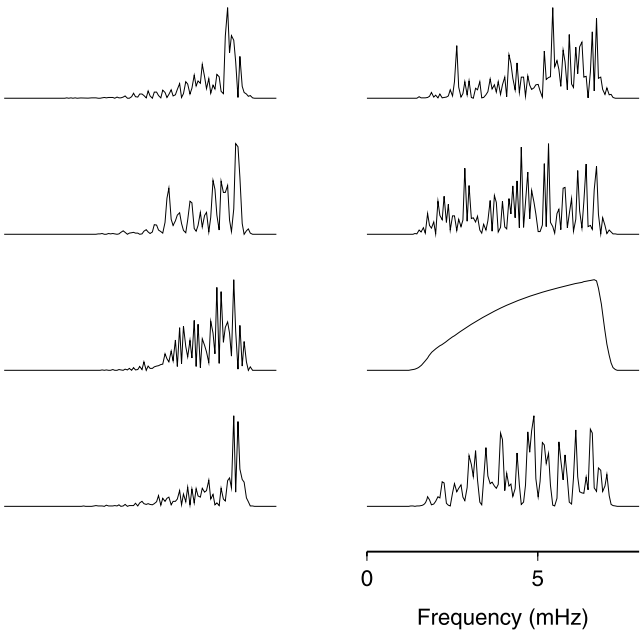

Figure 5. As Fig. 4, but for mantle-wave class: power spectra for the eight waveforms shown in Fig. 3.

learns to recognize patterns. Neural networks can learn to represent arbitrarily complex functions given only a set of input parameters and the corresponding outputs, allowing their use in cases where the details of this mapping are not understood well enough to allow the implementation of more conventional models.

In geophysics, neural networks have been applied to a number of problems: van der Baan & Jutten (2000) provide an introduction to some of these. Amongst others, they have been used to implement non-linear inversion schemes (Meier *et al.* 2007), and for identification of structural features in exploration seismology (Tingdahl & de Rooij 2005), along with the automatic picking of particular phase arrivals (Dai & MacBeth 1995, 1997; Wang 2002; Gentili &

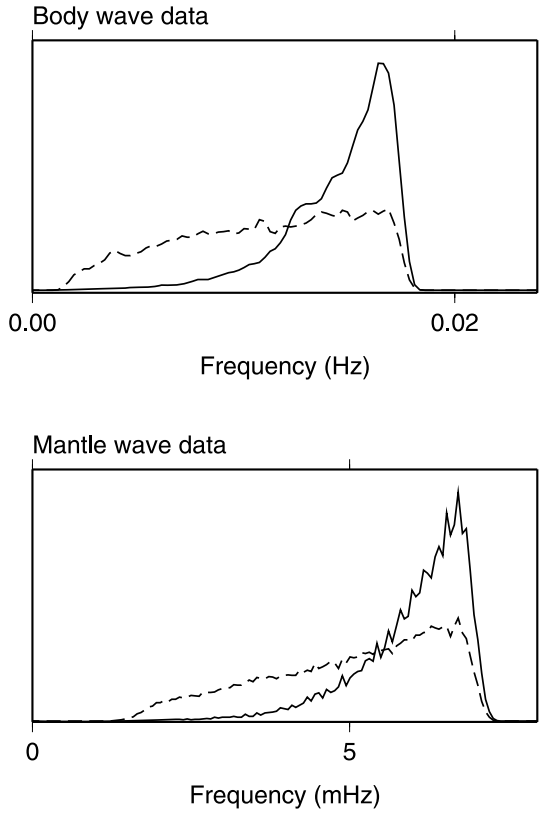

Figure 6. Average spectra for a range of body (top panel) and mantle (bottom panel) waveforms. The solid line corresponds to waveforms adjudged to be plausible; dashed line gives average of noisy waveforms. All spectra are normalized so that $\int_0^\infty |f(\omega)|^2 \, \mathrm{d}\omega = 1$ before summation.

Micheli 2006). Shimshoni & Intrator (1998) used neural networks to attempt to distinguish between seismograms of natural origin, and those with man-made sources; similarly, Scarpetta *et al.* (2005) use them to identify the origin of seismic events detected in the vicinity of Mount Vesuvius. However, we are not aware of any previous attempt to use them to assess the quality of seismic waveforms.

The theory of neural networks is an active research topic, and we will not attempt to provide more than an overview here—much literature is available if more detail is required (e.g. Bishop 1995; Mackay 2003). We consider only one form of network, which, as we see, gives satisfactory results. Many other approaches may exist, and we have not explored their behaviour or properties.

3.1 The single neuron

The basic building block of a neural network is the neuron. This represents a unit that takes a number of inputs, and returns a single output computed according to some known function. The sensitivity of the neuron to each input parameter is governed by a set of internal weights. Typically, the neuron will compute a weighted sum of all the inputs, termed the neuron's *activation*; the output is then some function of the activation. We can write the output of a neuron as

$$y = f(a)\,, \tag{1}$$

for a function, f. The activation, a, is computed

$$a = \sum_i^N w_i x_i\,, \tag{2}$$

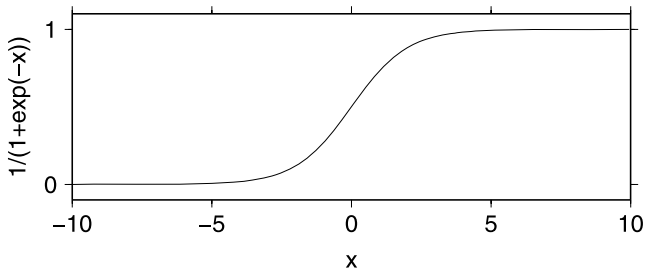

Figure 7. The logistic sigmoid function (eq. 4).

for some set of N inputs, $(x_1, x_2, \ldots, x_N)$, and their corresponding weights, $(w_1, w_2, \ldots, w_N)$.

The exact behaviour of the neuron is governed by the details of the 'activation function', f. A number of different functions are commonly used. For example, a linear neuron implements

$$f(a) = a\,. \tag{3}$$

We also make use of the sigmoid neuron, with activation function

$$f(a) = \frac{1}{1+\mathrm{e}^{-a}}\,. \tag{4}$$

This function, which is plotted in Fig. 7, has the property that

$$f'(a) = f(a)[1 - f(a)]. \tag{5}$$

which may prove useful in the implementation of a training rule.

It is possible to use a single neuron as a classifier, and this may be sufficient for simple problems. However, to recognize more complex patterns in data, we can connect multiple neurons together to form a network. Numerous network architectures exist, with particular properties and applications; for our purposes, one of the most common—the multilayer perceptron—will suffice.

3.2 Networks of neurons

The network topology to be used in this paper is depicted in Fig. 8. There are two distinct 'layers' of neurons: the network inputs are sent to each neuron in the first, 'hidden' layer. The outputs from the neurons in the hidden layer are used as the inputs for the neurons in the 'output' layer, and the outputs from these are taken to be the outputs for the network as a whole. Each layer incorporates a bias: one input to each neuron is a unit constant, which the neuron treats in the same manner as all other inputs. Note that the inputs to each neuron in a given layer are identical: however, as each neuron has its own set of weights, it handles the inputs in a unique fashion, and neurons in a given layer will produce different outputs. Typically, we start by randomizing all weights in the network, so that neurons have sensitivity to different aspects of the data.

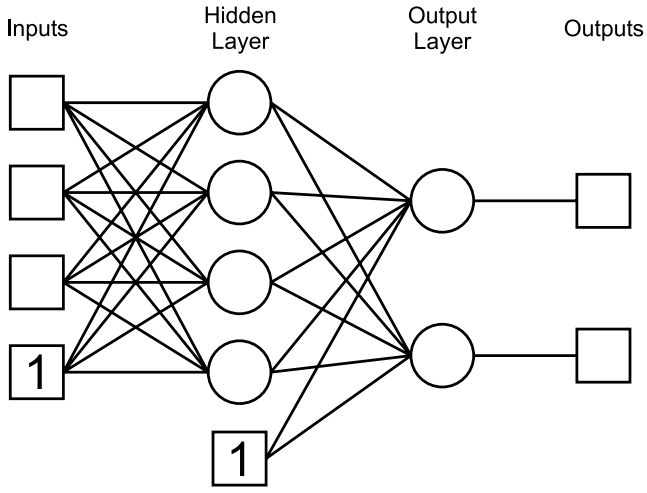

Figure 8. Schematic showing typical architecture for neural network with one hidden layer. The network takes three inputs and returns two outputs; circles represent neurons and squares the input and output parameters. Squares containing the number '1' represent bias inputs (see text).

The implementation of such a network is straightforward. Operation of the network proceeds on a layer-by-layer basis: given a set of inputs, each neuron in the first layer computes its output; then each neuron in the second layer, and so on. For a two-layer network as depicted in Fig. 8, we denote the output from the jth neuron in the hidden layer by y_j, computed

$$a_j = \sum_{i=1}^{N} w_{ji} x_i \tag{6}$$

$$y_j = f_j(a_j)\,, \tag{7}$$

for some activation function f_j. Similarly, the kth neuron in the output layer has output

$$b_k = \sum_{j=1}^{M} v_{kj} y_j \tag{8}$$

$$z_k = g_k(b_k)\,. \tag{9}$$

Here, v_{kj} represents the weights associated with the M inputs to the kth neuron.

Suppose some input vector, $\mathbf{x}^{(1)}$ is associated with an L-element desired output vector $\mathbf{d}^{(1)}$. Using eqs (7) and (9), we can compute the actual output from the network, $\mathbf{z}^{(1)}$. We then choose to define the error in the output from the network to be the sum-of-squares difference between desired and obtained values

$$E = \frac{1}{2}\sum_{k=1}^{L}\left(z_k^{(1)} - d_k^{(1)}\right)^2\,. \tag{10}$$

Computing partial derivatives with respect to the neuron weights, and defining

$$\Delta_k = \left(z_k^{(1)} - d_k^{(1)}\right) g_k'(b_k) \tag{11}$$

we obtain

$$\frac{\partial E}{\partial v_{kj}} = \Delta_k y_j \tag{12}$$

and

$$\frac{\partial E}{\partial w_{ji}} = x_i^{(1)} f_j'(a_j) \sum_{k=1}^{L} \Delta_k v_{kj}\,. \tag{13}$$

We can therefore reduce the overall error in the outputs by making the adjustments

$$v_{kj} \rightarrow v_{kj} - \eta \Delta_k y_j \tag{14}$$

$$w_{ji} \rightarrow w_{ji} - \eta x_i^{(1)} f_j'(a_j) \sum_{k=1}^{L} \Delta_k v_{kj}\,, \tag{15}$$

where η represents the 'learning rate'. This is a small positive number, ($\eta < 1$), and may be chosen to decrease as training progresses. The simple form of eqs (14) and (15) arise from the choice of error function in eq. (10). Other valid choices exist, and these will alter the training behaviour—and thus, the operation—of the networks.

3.3 Multiple networks

Since the initial weights are random, networks of identical architecture trained on the same data may not produce identical outputs when presented with a given set of inputs. Consequently, some networks may give better performance than others, at least for some subset of the data. It is therefore common to make use of 'committees' of networks, operating independently from one another. By averaging the outputs of all the networks, we improve our confidence in the classification—just as repeated measurement of some physical quantity reduces the random error associated with the measurement.

4 CLASSIFICATION OF SEISMIC WAVEFORMS

Even with the restriction to one particular class of network, it is clear that there are numerous choices to be made when implementing a selection algorithm. Taking into account that other classes of network exist that may be applied to this problem, the results presented here should be treated as a 'proof-of-concept', and not as a statement of the optimal classification method. Nevertheless, the choices presented here result in a system that performs adequately for practical purposes—as we demonstrate.

4.1 Network architecture

It is desirable to keep the number of input parameters to a network as small as possible—clearly, each additional input leads to a large number of additional weights in the network, and increases the amount of training that may be required to obtain satisfactory performance. Obviously, this must be balanced against the need to preserve sufficient features in our data set to enable classification. Using the entire discretized power spectrum is therefore unnecessary; an N-point representation of it is sufficient, for some modest N. To obtain such a representation, we define

$$P_n = \frac{1}{\alpha \delta_\omega} \int_{\omega_a+(n-1)\delta_\omega}^{\omega_a+n\delta_\omega} |f(\omega)|^2 \, \mathrm{d}\omega \quad (n = 1, 2, \ldots, N), \tag{16}$$

where ω_a and $\omega_b (\omega_a < \omega_b)$ represent the outer limits of the bandpass filter (i.e. the range in which the spectrum may be non-zero), and

$$\alpha = \int_{\omega_a}^{\omega_b} |f(\omega)|^2 \, \mathrm{d}\omega, \tag{17}$$

$$\delta_\omega = \frac{\omega_b - \omega_a}{N}. \tag{18}$$

Empirically, we find $N = 15$ provides adequate performance for our purposes. Since we know that event depth and source–receiver distance may each strongly influence the character of the waveform, we include additional parameters

$$P_\Delta = \frac{\Delta}{180} \tag{19}$$

$$P_z = \frac{z}{660} \tag{20}$$

for epicentral angle Δ (measured in degrees) and event depth z. These parameters are chosen so that all values lie in the range $(0, 1)$, to ensure that all parameters have similar magnitude. As noted earlier, this additional information is readily available to an accuracy that is more than sufficient for our purposes.

These inputs are fed into a network of the form depicted in Fig. 8. We use separate networks for each class of waveform to be analysed, although all networks have identical structure. We use 60 neurons in the hidden layer, each with a logistic sigmoid activation function; the output layer contains two neurons, with linear activation functions (Bishop 1995). The network therefore yields a two-element output vector: we take $(1, 0)$ to correspond to acceptable data, and $(0, 1)$ to represent unusable waveforms. As set out in Section 3.3, it is beneficial to train multiple networks simultaneously; we therefore create ten networks for each class. All neuron weights are initially random on a uniform distribution in the range $(-1, 1)$: this is the only difference between networks prior to the commencement of training.

4.2 Data sets for training and testing

To train and assess the performance of our network, we must create two data sets, and classify the waveforms by hand. The first will be used to optimize the weights in the networks; the second data set is required to provide an independent measure of how successfully the optimized networks perform their task. Obviously, we require both data sets to be broadly representative of the data the network is ultimately to classify.

To this end, we assemble a global data set containing seismograms from global IRIS/IDA stations, for 65 events of magnitude 6+ occurring in 2000. We preprocess all traces, to ensure that:

(i) all records are correctly formed, with all necessary information present and within sensible limits,
(ii) excessively short traces are ignored and
(iii) amplitudes are within physical limits.

Traces that fail to meet these criteria are discarded. All traces are rotated to provide three components, aligned with vertical, north–south and east–west axes. We then select two sets of 1000 traces each, at random and without replacement, from across all events. The two classes of waveform described in Section 2 are then extracted from each seismogram, and assessed visually to be either plausible or noisy. Again, we emphasize the subjective nature of this classification, and that we do not necessarily expect an automated system to agree with the visual assessment in every single case.

We then use the first classified data set to run the training algorithm, as set out in Section 3.2. We use a learning rate parameter that decreases over time: for the ith example,

$$\eta_i = \max(\eta_0 \mathrm{e}^{-\frac{i}{\tau}}, \epsilon), \tag{21}$$

where ϵ is included to enforce a minimum learning rate. For the examples presented here, we choose

$$\begin{aligned} \eta_0^{\mathrm{out}} &= 0.2 \quad & \eta_0^{\mathrm{hid}} &= 0.5 \\ \tau^{\mathrm{out}} &= 500 \quad & \tau^{\mathrm{hid}} &= 2000 \\ \epsilon &= 0.05, \end{aligned} \tag{22}$$

where η_0^{hid} and τ^{hid} are used when updating the hidden layer weights via eq. (15). Updates to the output layer weights, performed according to eq. (14), are controlled by η_0^{out} and τ^{out}. We find that the overall performance of the networks appears stable under modest changes in these settings.

In practice, we find that the predominance of poor-quality data in our data set causes difficulties during training: whilst a reasonable separation between classes can still be obtained, all outputs are biased towards those for poor data. To correct for this, we weight

the learning rate at each training step according to the Shannon information content of the known classification, defined

$$h(x) = \log_2 \frac{1}{P(x)}, \tag{23}$$

where $P(x)$ is the *a priori* probability of obtaining outcome x estimated from the relative frequency of each class in the data encountered by the network thus far (Mackay 2003). Our learning rate parameter therefore becomes

$$\eta_i = \max(\eta_0 \mathrm{e}^{-\frac{i}{\tau}}, \epsilon) \cdot \left\{ -\log_2 [P(\text{good})] \right\} \tag{24}$$

when the update is made using information from a good-quality trace, and

$$\begin{aligned} \eta_i &= \max(\eta_0 \mathrm{e}^{-\frac{i}{\tau}}, \epsilon) \cdot \left\{ -\log_2 [P(\text{bad})] \right\} \\ &= \max(\eta_0 \mathrm{e}^{-\frac{i}{\tau}}, \epsilon) \cdot \left\{ -\log_2 [1 - P(\text{good})] \right\} \end{aligned} \tag{25}$$

when the update arises from a poor-quality trace. This causes our algorithm to learn more from less common outcomes—which is a desirable characteristic. However, it does potentially increase the size of step taken when updating our weights, which under some circumstances may cause problems. An alternative approach to this problem might entail creating a reduced training data set by selecting an equal number of good and bad traces from our initial data set, although this would require the entire training data set to be known before training commences—whereas the training algorithm outlined here may be used for continuous learning.

4.3 Effectiveness

The trained networks may now be applied to the second data set, which consists entirely of traces that the network has not previously 'seen'. For each, the network generates a vector (x, y); this may then be interpreted to decide whether a particular waveform should be regarded as plausible. Fig. 9 shows the distribution of output vectors we obtain, separated according to our visual classification. We see that there is a reasonable separation between the classes, with traces classified as good being strongly clustered around the point (1, 0). As expected, the broad definition of a poor-quality trace leads to the outputs arising from these being less clustered, although there is still a strong bias towards the point (0, 1).

To make use of the neural network for classification, we must convert the network output vectors into a classification. To do this, we compute a quality score, defined to be the Euclidean distance between a particular trace's output vector (x, y), and the 'ideal' bad trace, (0, 1)

$$Q = \sqrt{x^2 + (y-1)^2} \tag{26}$$

and define a threshold below which a trace will be declared to be bad. We note that this metric is closely related to the error function used in training the network; more sophisticated approaches derived from the statistical properties of the network are also possible, and may yield improved results. From Fig. 9, we see that it will be impossible to completely differentiate the two classes—as stated earlier. Choosing the optimal threshold therefore involves a trade-off between the number of bad traces remaining in the 'cleaned' data set, and the number of good traces removed unnecessarily. To illustrate this, Fig. 10 shows the percentage of each remaining in the data set, as the threshold distance is increased. We see that, in broad terms, it is possible to remove around 60 per cent of bad traces for minimal loss of good data; if we are prepared to sacrifice 10 per cent of plausible traces, we can eliminate 90 per cent of bad waveforms. Obviously, the precise figures will vary between data sets; our demands may also change depending on the intended

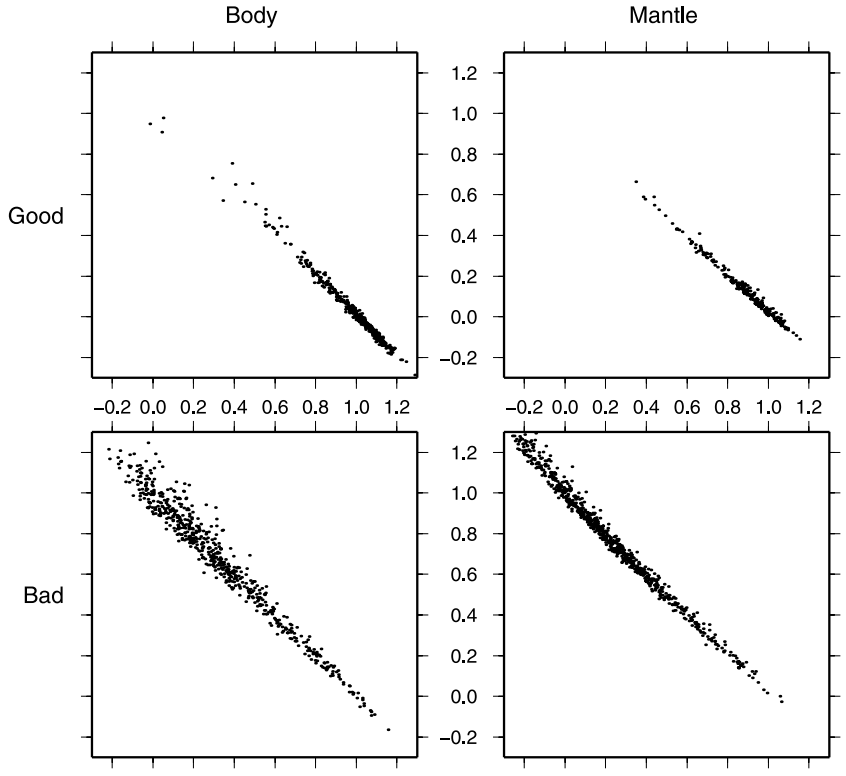

Figure 9. Output vectors, (x, y), generated by neural networks for waveforms in second (test) data set. Outputs have been separated according to visual classification (information not available to network) and according to waveform class. Point (1, 0) represents good data; (0, 1) represents bad data.

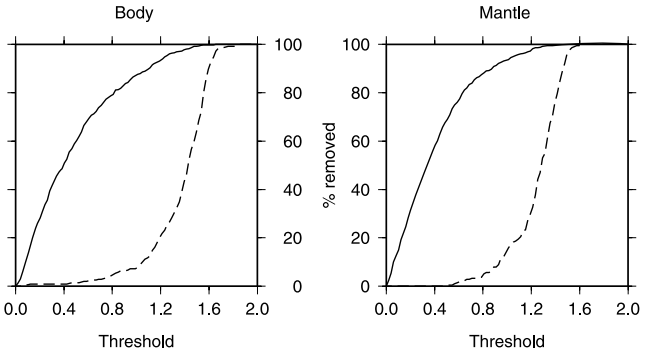

Figure 10. Percentage of visually bad (solid line) and good (dashed line) traces removed from validation data set for given threshold.

uses of the cleaned data set. We find that changing the network parameters from those set out in eq. (22) does not significantly change the relationship between the amount of bad data retained, and good data lost, although the threshold at which a given percentage occurs may alter somewhat.

5 DEMONSTRATION: AUTOMATIC MOMENT TENSOR DETERMINATION

To illustrate the relationship between visual quality and the network-assigned quality score, we obtain seismograms from global networks for the event 010800G. This is pre-processed, body- and mantle-wave windows are selected, and then classified by the trained networks. Fig. 11 shows a representative selection of body wave traces, arranged in increasing order of network-assigned quality; Fig. 12 shows mantle-wave traces in the same fashion. Synthetic traces are also shown, to inform a visual assessment, although some differences between these and the data are to be expected. It is clear that the computed score correlates reasonably well with a visual assessment, as is required if the system is to have practical use.

We note that the relationship between numerical score and visual appearance of the trace varies between classes, and thus an appropriate threshold value may also vary. Where this threshold should be drawn depends to a large extent on the intended application of the cleaned data set; however, a threshold of around 1.0 would appear reasonable for body wave traces, and a somewhat higher cut-off for mantle-wave traces.

To demonstrate a practical application using the cleaned data sets, we can perform CMT-style inversion for event source parameters and spatio-temporal location (Dziewonski *et al.* 1981; Dziewonski & Woodhouse 1983; Valentine & Woodhouse 2010). The process is fully automatic: once a data set has been obtained from (for example) the IRIS data centre, it is windowed and inverted without human intervention. Rapid automatic computation of event mechanisms is an important task for earthquake hazard assessment systems, and has been considered from a number of angles (e.g. Tajima *et al.* 2002; Bernardi *et al.* 2004; Clinton *et al.* 2006), although usually on a regional scale, since detection of smaller events at teleseismic distances can be poor.

We perform data selection and inversion using a range of threshold values, to demonstrate the relationship between data set quality and the results obtained. For simplicity, we use the same threshold for both body- and mantle-wave classes. Results for a variety of events are presented in Table 1, with the Global CMT Catalogue solution for reference.

With the exception of the results for event 062100A, which we discuss in more detail below, we see that the network-based data selection method is able to produce source parameters in good agreement with the catalogue value. We note that we do not expect exact agreement: the details of our inversion algorithm will differ, as will the earth models used. For this reason, it is difficult to quantify exactly the degree of agreement; calculation of realistic errors on source parameters is not straightforward, and those given in catalogues are generally accepted to significantly underestimate the uncertainties involved (see also Valentine & Woodhouse 2010). One measure of the relative quality of solution is provided by the overall waveform misfit, defined

$$m^2 = \frac{(\mathbf{d}-\mathbf{s})^{\mathbf{T}}(\mathbf{d}-\mathbf{s})}{\mathbf{d}^{\mathbf{T}}\mathbf{d}} \tag{27}$$

which the inversion process attempts to minimize. Here, $\mathbf{d}$ represents the vector of all data waveforms; $\mathbf{s}$ is the corresponding vector of synthetic waveforms computed using the best available source. When performing source inversions using hand-picked data, a waveform misfit of less than 0.5 is generally considered reasonable; obtaining a misfit below 0.2 would be exceptional. Of course, a slight reduction in misfit is to be expected as we discard traces—our solution has to satisfy fewer constraints—but the improvements seen as the overall data set quality increases are significant. A threshold choice in the range 1.0–1.2 results in a data set of reasonable size, but which can be satisfactorily explained by a seismic source; this is in accordance with our earlier comments linking numerical and visual quality.

The exception to this appears to be event 062100A, for which we have been unable to recover a good-quality source. An inspection of the classified waveforms for this event highlights a potential problem with the current network-based approach: since we use only normalized spectra, we are not always able to detect pure amplitude errors in our data set—that is, cases where the overall form of the trace is reasonable, but where inaccurate instrument response information leads to a uniform scaling of all points in the trace. Many errors of this type result in sufficiently high amplitudes that the offending traces are rejected at the pre-processing stage, but some may remain. Unfortunately, the source inversion process is highly sensitive to such errors, and they must be trapped before the data set is used. This is a relatively straightforward process, involving discarding any traces with a maximum amplitude significantly higher than others in the data set for that event. Introducing this as a post-processing stage after network-based classification, we are able to automatically obtain the source parameters listed in Table 2, which show a significant improvement over the original determination.

As this experience dictates, it is important to develop techniques for monitoring the performance of the network-based classification system, and highlighting any unusual behaviour for further investigation. At a most basic level, it is advisable to visually inspect some subset of the network classifications. It is also straightforward to monitor the classification statistics to identify any events which appear to be handled in an unusual fashion. If synthetics are available for a given event, these may themselves be submitted to the network for classification. Obviously, we expect that they will all be assessed to be 'good': if this fails for any significant number of traces, the classification of real data is unlikely to be robust.

Within an application of the classified data set, further refinement may be possible. For the current case of moment tensor determination, it is straightforward to compute the contribution each trace makes to the waveform misfit. It is sometimes found that elimination of one or two traces will result in a significant improvement in

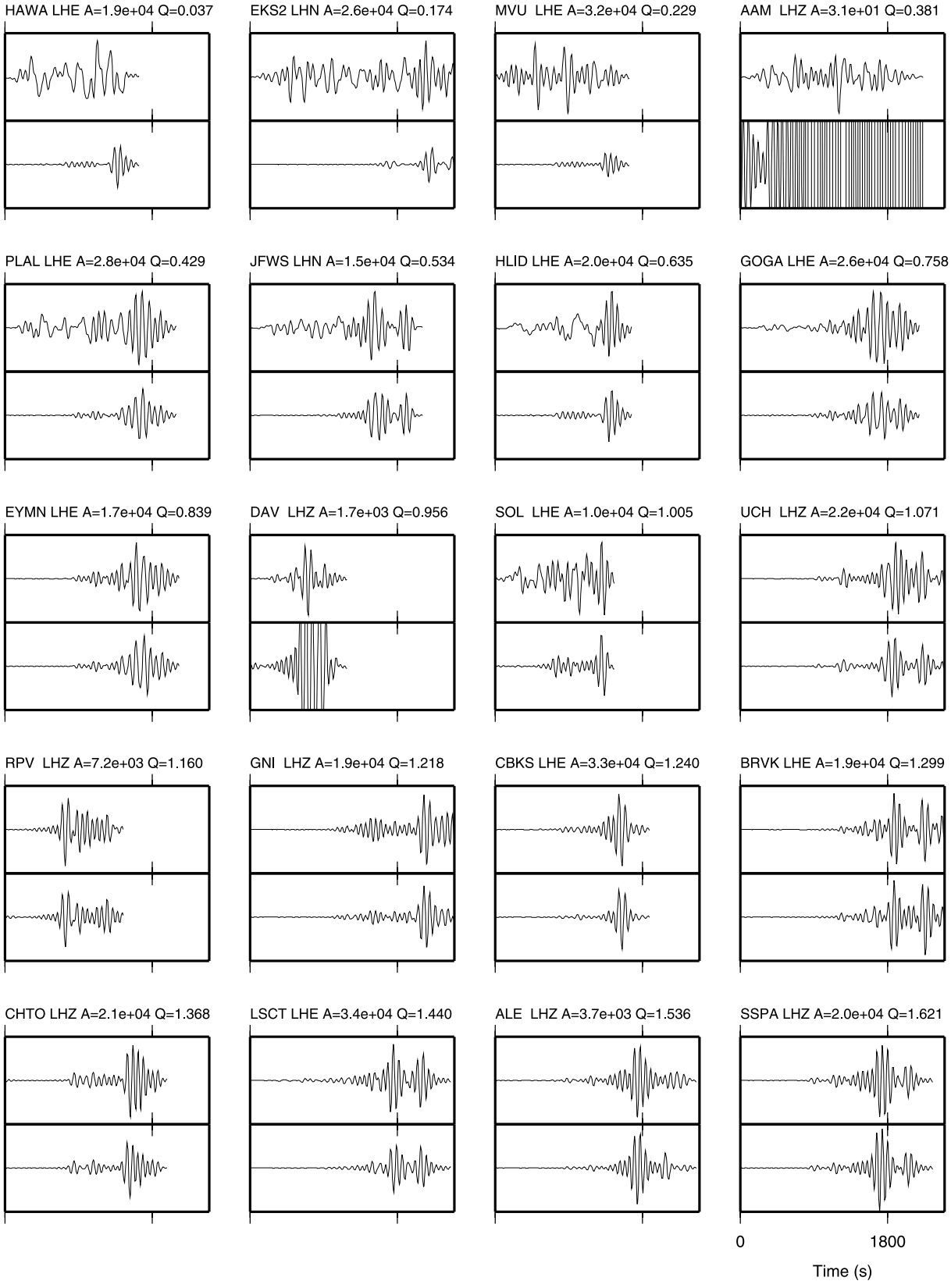

Figure 11. Sample body-wave traces (top panels) and corresponding synthetics (bottom panels) from event 010800G, arranged in ascending order of network-assigned quality factor (Q), defined as in eq. (26). All traces are scaled by the maximum data amplitude (A); synthetics are plotted on same scale as the data. Time scales in all panels are identical.

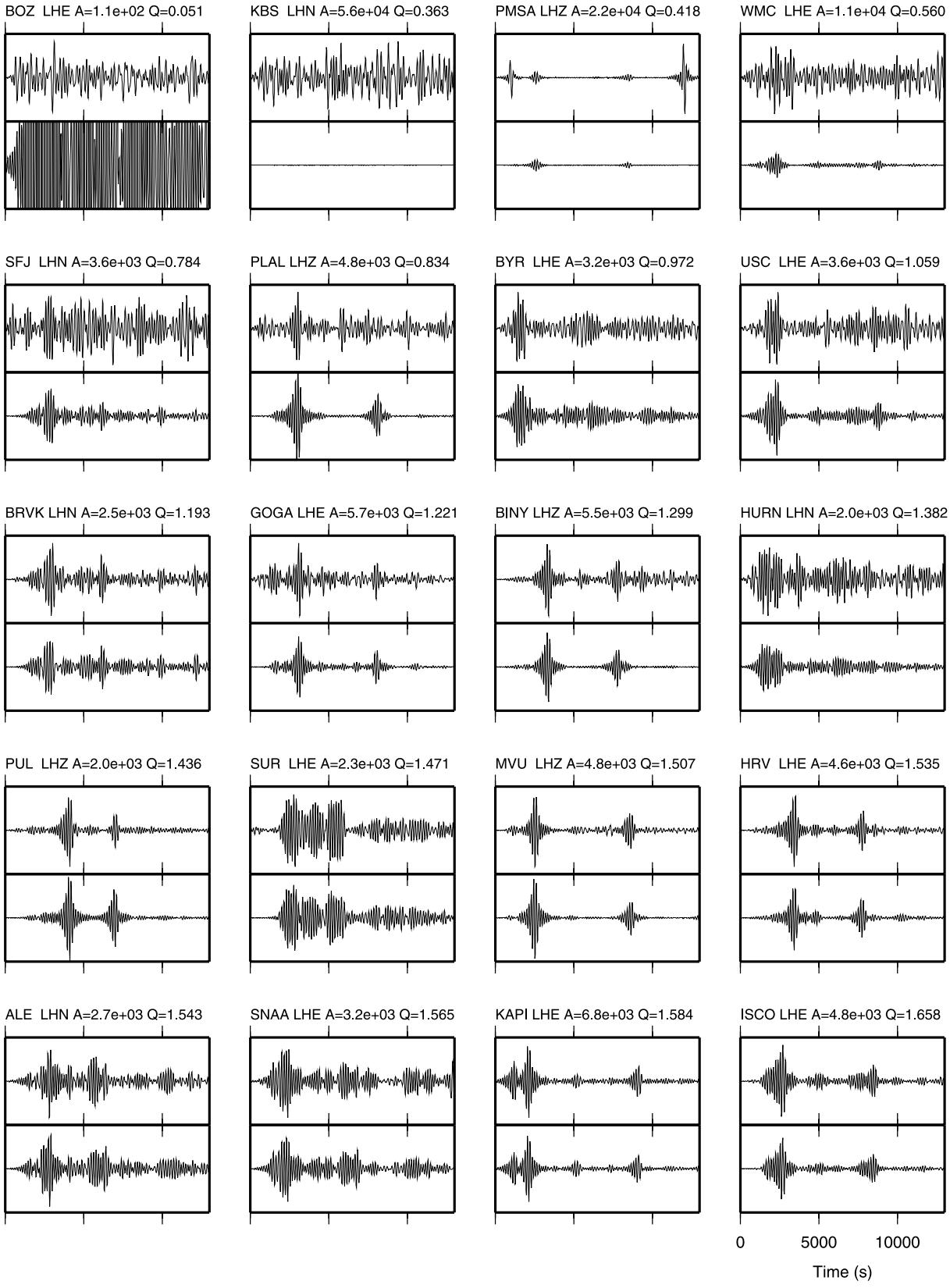

Figure 12. Sample mantle-wave traces (top panels) and corresponding synthetics (bottom panels) from event 010800G, arranged in ascending order of network-assigned quality factor (Q), defined as in eq. (26). All traces are scaled by the maximum data amplitude (A); synthetics are plotted on same scale as the data. Timescales in all panels are identical.

Table 1. Source parameters recovered by source inversion of automatically selected data sets for given events.

Event	Source	N_B	N_M	M_{rr}	$M_{\theta\theta}$	$M_{\phi\phi}$	$M_{r\theta}$	$M_{r\phi}$	$M_{\theta\phi}$	Exp	Lat	Lon	Dep	Time	Misfit
010800G	CMT			−0.12	−0.60	0.72	0.37	6.78	1.01	26	−16.84	−173.81	162.4	16:47:30.2	
	0.4	325	296	0.04	−0.72	0.68	0.30	6.23	0.59	26	−16.83	−173.98	160.0	16:47:31.4	0.665
	0.8	234	251	0.02	−0.66	0.64	0.37	6.24	0.64	26	−16.84	−173.98	159.5	16:47:31.4	0.418
	1.2	115	187	−0.02	−0.54	0.57	0.25	6.32	0.76	26	−16.83	−173.98	156.2	16:47:31.4	0.297
	1.4	44	67	−0.03	−0.68	0.71	0.25	6.63	0.53	26	−16.90	−174.00	155.7	16:47:30.4	0.205
020600C	CMT			6.65	−4.72	−1.92	6.00	2.87	−3.10	25	−6.06	151.19	45.0F	11:34:01.7	
	0.4	386	281	0.42	−0.51	0.09	2.15	0.52	−0.14	26	−5.85	151.18	15.0F	11:34:02.5	0.993
	0.8	300	157	5.60	−3.96	−1.65	5.72	2.90	−2.72	25	−6.05	151.30	49.8	11:34:02.9	0.964
	1.2	234	86	6.19	−4.27	−1.92	5.71	2.96	−2.45	25	−6.02	151.17	46.3	11:34:02.5	0.346
	1.4	175	15	5.08	−4.11	−0.97	5.57	4.14	−3.51	25	−5.92	151.12	41.2	11:34:01.8	0.227
042300B	CMT			−0.22	0.04	0.18	−0.49	3.03	−0.25	26	−28.41	−63.04	607.9	09:27:29.4	
	0.4	533	300	−0.16	−0.02	0.17	−0.24	2.89	−0.07	26	−28.42	−62.97	614.8	09:27:29.5	0.993
	0.8	444	97	0.03	−0.19	0.16	0.12	2.94	0.35	26	−28.46	−63.00	612.8	09:27:28.8	0.997
	1.2	341	22	0.64	−0.25	−0.39	−0.44	2.51	−0.37	26	−28.60	−62.65	612.4	09:27:27.9	0.905
	1.4	225	0	−0.07	0.15	−0.08	−0.51	3.10	−0.08	26	−28.15	−63.06	612.4	09:27:31.0	0.354
062100A	CMT			−0.75	0.60	0.15	0.20	−1.14	5.30	25	63.98	−20.85	15.0F	00:51:54.8	
	0.4	344	376	−6.86	3.30	3.55	−3.13	0.66	9.52	25	64.43	−21.16	15.0F	00:51:57.7	0.998
	0.8	217	219	−0.30	0.08	0.21	1.45	1.30	0.84	26	64.26	−21.23	15.0F	00:51:53.4	0.987
	1.2	127	136	−0.40	0.42	−0.03	1.11	0.45	0.77	26	64.28	−20.45	15.0F	00:51:51.1	0.984
	1.4	62	69	−4.75	2.88	1.86	−0.71	−1.83	8.69	25	63.74	−21.77	15.0F	00:51:51.8	0.891
080400G	CMT			1.51	0.20	−1.72	−0.30	−0.86	0.50	26	48.77	142.03	15.0F	21:13:12.1	
	0.4	314	324	1.39	0.23	−1.62	−0.12	−0.83	0.54	26	48.88	142.22	15.0F	21:13:14.8	0.980
	0.8	236	227	1.51	0.21	−1.70	−0.25	−0.57	0.58	26	48.78	142.16	15.0F	21:13:14.7	0.796
	1.2	166	160	1.49	0.18	−1.67	0.07	−0.92	0.61	26	48.84	142.12	15.0F	21:13:16.0	0.209
	1.4	116	70	1.59	0.17	−1.77	0.31	−1.16	0.68	26	48.76	142.15	15.0F	21:13:15.9	0.167

Notes: Column headed 'source' gives threshold used in data selection, or 'CMT' where CMT catalogue values are stated. N_B is the number of body waveforms used in inversion, and N_M refers to mantle waveforms. M_{ij} represent the six independent components of the moment tensor, in units of 10^{Exp} dyne cm. An 'F' appearing in the depth column denotes that this component was fixed during the course of inversion. Misfit is defined as in eq. (27).

Table 2. The benefits of post-processing.

Event	Source	N_B	N_M	M_{rr}	$M_{\theta\theta}$	$M_{\phi\phi}$	$M_{r\theta}$	$M_{r\phi}$	$M_{\theta\phi}$	Exp	Lat	Lon	Dep	Time	Misfit
042300B	0.8a	442	95	−0.08	0.13	−0.04	−0.26	3.12	−0.21	26	−28.46	−62.95	618.0	09:27:29.1	0.949
	0.8ab	442	93	−0.10	0.13	−0.03	−0.36	3.04	−0.23	26	−28.42	−62.87	618.6	09:27:29.2	0.892
	1.2a	340	21	−0.34	0.10	0.23	−0.38	2.29	−0.17	26	−28.41	−62.89	621.8	09:27:29.9	0.500
	1.2ab	340	20	−0.27	0.08	0.19	−0.45	2.67	−0.17	26	−28.41	−62.84	619.7	09:27:29.0	0.272
062100A	0.8a	207	201	−1.17	0.91	0.25	0.68	−1.72	5.17	25	64.01	−21.07	15.0F	00:51:55.0	0.682
	1.2a	122	131	−1.25	1.15	0.10	−0.48	−1.29	5.14	25	63.95	−21.05	15.0F	00:51:53.2	0.466
	1.2ab	122	130	−1.22	1.17	0.05	−0.33	−1.37	5.18	25	63.99	−20.99	15.0F	00:51:52.4	0.359
	1.4a	62	67	−1.23	0.79	0.45	−0.35	−4.26	5.22	25	63.87	−21.31	15.0F	00:51:52.5	0.540
080400G	0.8a	234	227	1.43	0.25	−1.68	−0.15	−0.71	0.57	26	48.78	142.15	15.0F	21:13:15.2	0.508

Notes: Source parameters recovered by source inversion of selected data sets from Table 1 with additional cleanup stages: (a) Removal of anomalously high amplitude traces; (b) Removal of traces contributing disproportionately to misfit. All quantities are defined as in Table 1. Note that we only show results where the post-processing stages resulted in removal of traces: results from events 010800G and 020600C remain as in Table 1.

the fit—visual inspection can then determine whether these should be removed from the data set. An example is shown in Table 2, for event 062100A, where removal of one trace results in a 20 per cent improvement in misfit.

6 CONCLUSIONS

We have demonstrated that neural networks can be trained to assign a quality score to seismic waveforms, and that this provides a robust method for removing noisy traces from a real data set. At present, the network uses only the spectral representation of the waveform, an estimate of event depth, and an estimate of the source–receiver distance; as a result, the classification is made independent from any assumptions about earth structure, or the physics of wave propagation. This allows us to be confident that the resulting data set has no *a priori* bias that may affect, for example, tomographic inversion.

We do not present this system as a 'finished product': clearly, there is much scope for development and enhancement. It may be possible to identify parameters beyond the basic spectral representation that will improve our ability to distinguish good and bad traces; changes to the network architecture may be beneficial. Consideration should also be given to how we might best use the information gained from the network classification—for example, it may be possible to incorporate quality information into the weighting scheme used in a tomographic inversion. However, the system as set out in this paper functions sufficiently well for practical purposes.

The performance of a neural network depends crucially on the data set used for training purposes: the network cannot be expected to perform well when presented with data it has not learnt to use. The networks demonstrated here were trained on events of magnitude 6+; they could not then be used to classify seismograms from significantly smaller events—although appropriately trained networks could be generated. Additionally, any model derived

from data will always exhibit some signature of the data selection strategy; automating the selection process does not remove the need to carefully consider the relationship between data set and results, and the trade-off between data quantity and data quality. This assessment may vary considerably between different applications.

Our development of this automatic data selection system has been motivated by the need to handle large quantities of data when performing global seismic tomography, in a well-defined and repeatable manner. However, the need for high-quality data sets extends to other areas of seismology, and the current system may be adapted to suit these applications. In particular, the ability to rapidly identify good-quality waveforms may be beneficial in systems relying on real-time analysis of seismic data in relation to earthquake and tsunami hazards.

ACKNOWLEDGMENTS

The authors are grateful to two anonymous reviewers for their comments and suggestions. APV is supported by the UK Natural Environment Research Council under the grant NE/C510916/1 and by Worcester College, Oxford. Computational and infrastructural support has also been provided through NE/B505997/1.

REFERENCES

Allen, R., 1978. Automatic earthquake recognition and timing from single traces, *Bull. seism. Soc. Am.,* **68,** 1521–1532.

Baer, M. & Kradolfer, U., 1987. An automatic phase picker for local and teleseismic events, *Bull. seism. Soc. Am.,* **77,** 1437–1445.

Bernardi, F., Braunmiller, J., Kradolfer, U. & Giardini, D., 2004. Automatic regional moment tensor inversion in the European-Mediterranean region, *Geophys. J. Int.,* **157,** 703–716.

Bishop, C., 1995. *Neural Networks for Pattern Recognition,* Oxford University Press, Oxford.

Bolt, B., 1960. The revision of earthquake epicentres, focal depths and origin-times using a high-speed computer, *Geophys. J. R. astr. Soc.,* **3,** 433–440.

Clinton, J., Hauksson, E. & Solanki, K., 2006. An evaluation of SCSN moment tensor solutions: robustness of the m_w magnitude scale, style of faulting, and automation of the method, *Bull. seism. Soc. Am.,* **96,** 1689–1705.

Dai, H. & MacBeth, C., 1995. Automatic picking of seismic arrivals in local earthquake data using an artificial neural network, *Geophys. J. Int.,* **120,** 758–774.

Dai, H. & MacBeth, C., 1997. The application of back-propagation neural network to automatic picking seismic arrivals from single-component recordings, *J. geophys. Res.,* **102,** 15 105–15 113.

Di Stefano, R., Aldersons, F., Kissling, E., Baccheschi, P. & Chiarabba, C., 2006. Automatic seismic phase picking and consistent observation error assessment: application to the Italian seismicity, *Geophys. J. Int.,* **165,** 121–134.

Dziewonski, A. & Woodhouse, J., 1983. An experiment in the systematic study of global seismicity: centroid-moment tensor solutions for 201 moderate and large earthquakes of 1981, *J. geophys. Res.,* **88,** 3247–3271.

Dziewonski, A., Chou, T.-A. & Woodhouse, J., 1981. Determination of earthquake source parameters from waveform data for studies of global and regional seismicity, *J. geophys. Res.,* **86,** 2825–2852.

Gentili, S. & Micheli, A., 2006. Automatic picking of P and S phases using a neural tree, *J. Seismol.,* **10,** 39–63.

Lebedev, S. & van der Hilst, R., 2008. Global upper-mantle tomography with the automated multimode inversion of surface and S-wave forms, *Geophys. J. Int.,* **173,** 505–518.

Lebedev, S., Nolet, G., Meier, T. & van der Hilst, R., 2005. Automated multimode inversion of surface and S waveforms, *Geophys. J. Int.,* **162,** 951–964.

Mackay, D., 2003. *Information Theory, Inference and Learning Algorithms,* Cambridge University Press, Cambridge.

Maggi, A., Tape, C., Chen, M., Chao, D. & Tromp, J., 2009. An automated time-window selection algorithm for seismic tomography, *Geophys. J. Int.,* **178,** 257–281.

Meier, U., Curtis, A. & Trampert, J., 2007. Global crustal thickness from neural network inversion of surface wave data, *Geophys. J. Int.,* **169,** 706–722.

Pasyanos, M., Dreger, D. & Romanowicz, B., 1996. Towards real-time estimation of regional moment tensors, *Bull. seism. Soc. Am.,* **86,** 1255–1269.

Scarpetta, S., Giudicepietro, F., Ezin, E., Petrosino, S., Pezzo, E.D., Martini, M. & Marinaro, M., 2005. Automatic classification of seismic signals at Mt. Vezsuvius volcano, Italy, using neural networks, *Bull. seism. Soc. Am.,* **95,** 185–196.

Scognamiglio, L., Tinti, E. & Michelini, A., 2009. Real-time determination of seismic moment tensor for the Italian region, *Bull. seism. Soc. Am.,* **99,** 2223–2242.

Shimshoni, Y. & Intrator, N., 1998. Classification of seismic signals by integrating ensembles of neural networks, *IEEE Trans. Signal. Proc.,* **46,** 1194–1201.

Tajima, F., Mégnin, C., Dreger, D. & Romanowicz, B., 2002. Feasibility of real-time broadband waveform inversion for simultaneous moment tensor and centroid location determination, *Bull. seism. Soc. Am.,* **92**(2), 739–750.

Tape, C., Liu, Q., Maggi, A. & Tromp, J., 2009. Adjoint tomography of the southern California crust, *Science,* **325,** 988–992.

Thurber, C. & Ritsema, J., 2007. *Theory and Observations – Seismic Tomography and Inverse Methods,* Vol. 1, Chapter 10, pp. 323–360, Elsevier, Amsterdam.

Tingdahl, K. & de Rooij, M., 2005. Semi-automatic detection of faults in 3D seismic data, *Geophys. Prospect.,* **53,** 533–542.

Valentine, A.P. & Woodhouse, J.H., 2010. Reducing errors in seismic tomography: combined inversion for sources and structure, *Geophys. J. Int.,* **180**(2), 847–857.

van der Baan, M. & Jutten, C., 2000. Neural networks in geophysical applications, *Geophys.,* **65,** 1032–1047.

van Heijst, H. & Woodhouse, J., 1997. Measuring surface-wave overtone phase velocities using a mode-branch stripping technique, *Geophys. J. Int.,* **131,** 209–230.

van Heijst, H. & Woodhouse, J., 1999. Global high-resolution phase velocity distributions of overtone and fundamental-mode surface waves determined by mode branch stripping, *Geophys. J. Int.,* **137,** 601–620.

Wang, J., 2002. Adaptive training of neural networks for automatic seismic phase identification, *Pure appl. Geophys.,* **159,** 1021–1041.

Woodhouse, J. & Dziewonski, A., 1984. Mapping the upper mantle: three-dimensional modelling of earth structure by inversion of seismic waveforms., *J. geophys. Res.,* **89,** 5953–5986.