THE UNIVERSITY *of* EDINBURGH

Methods for Bayesian inversion of seismic data

Matthew James Walker

Thesis submitted for the degree of
Doctor of Philosophy
The University of Edinburgh
2014

Declaration

I certify that this thesis, and the work presented herein, is my own original composition. Where it draws on the work of others, this is acknowledged at the appropriate points in the text. This work has not been previously submitted for any other degree, or professional qualification. Chapters 3, 4 and 5 comprise work that has been accepted for publication in peer-reviewed scientific journals. These articles may be accessed at the following locations on-line:

Chapter 3: `iopscience.iop.org/0266-5611/30/6/065002/`
(doi: 10.1088/0266-5611/30/6/065002)

Chapter 4: `onlinelibrary.wiley.com/doi/10.1002/2014JB011010/abstract`
(doi: 10.1002/2014JB011010)

Chapter 5: `gji.oxfordjournals.org/content/198/1/342`
(doi: 10.1093/gji/ggu132)

Matthew James Walker

Lay summary

In many applications seismic data is used to infer the physical properties of the subsurface by using the process of seismic inversion. However multiple configurations of the subsurface physical properties may give rise to the same observed seismic data thus there is no unique solution to such a problem, but rather a set of possible solutions. Bayesian seismic inversion methods seek to assign probabilities to each possible solution given the observed data and any prior information which may be available about the subsurface. The assignment of probabilities to each possible solution is usually a computationally expensive task since typically there are a very large, if not infinite, number of possible solutions. This thesis describes a number of methods whose purpose is to overcome this limitation. Furthermore, the collation of prior information, from numerous and often highly subjective sources, into a format usable in such methods is a difficult problem. Thus this thesis also describes a method whose aim is to aid this process.

Abstract

The purpose of Bayesian seismic inversion is to combine information derived from seismic data and prior geological knowledge to determine a posterior probability distribution over parameters describing the elastic and geological properties of the subsurface. Typically the subsurface is modelled by a cellular grid model containing thousands or millions of cells within which these parameters are to be determined. Thus such inversions are computationally expensive due to the size of the parameter space (being proportional to the number of grid cells) over which the posterior is to be determined. Therefore, in practice approximations to Bayesian seismic inversion must be considered. A particular, existing approximate workflow is described in this thesis: the so-called two-stage inversion method explicitly splits the inversion problem into elastic and geological inversion stages. These two stages sequentially estimate the elastic parameters given the seismic data, and then the geological parameters given the elastic parameter estimates, respectively. In this thesis a number of methodologies are developed which enhance the accuracy of this approximate workflow.

To reduce computational cost, existing elastic inversion methods often incorporate only simplified prior information about the elastic parameters. Thus a method is introduced which transforms such results, obtained using prior information specified using only two-point geostatistics, into new estimates containing sophisticated multi-point geostatistical prior information. The method uses a so-called deep neural network, trained using only synthetic instances (or 'examples') of these two estimates, to apply this transformation. The method is shown to improve the resolution and accuracy (by comparison to well measurements) of elastic parameter estimates determined for a real hydrocarbon reservoir.

It has been shown previously that so-called mixture density network (MDN) inversion can be used to solve geological inversion analytically (and thus very rapidly

and efficiently) but only under certain assumptions about the geological prior distribution. A so-called prior replacement operation is developed here, which can be used to relax these requirements. It permits the efficient MDN method to be incorporated into general stochastic geological inversion methods which are free from the restrictive assumptions. Such methods rely on the use of Markov-chain Monte-Carlo (MCMC) sampling, which estimate the posterior (over the geological parameters) by producing a correlated chain of samples from it. It is shown that this approach can yield biased estimates of the posterior. Thus an alternative method which obtains a set of non-correlated samples from the posterior is developed, avoiding the possibility of bias in the estimate. The new method was tested on a synthetic geological inversion problem; its results compared favourably to those of Gibbs sampling (a MCMC method) on the same problem, which exhibited very significant bias.

The geological prior information used in seismic inversion can be derived from real images which bear similarity to the geology anticipated within the target region of the subsurface. Such so-called training images are not always available from which this information (in the form of geostatistics) may be extracted. In this case appropriate training images may be generated by geological experts. However, this process can be costly and difficult. Thus an elicitation method (based on a genetic algorithm) is developed here which obtains the appropriate geostatistics reliably and directly from a geological expert, without the need for training images. 12 experts were asked to use the algorithm (individually) to determine the appropriate geostatistics for a physical (target) geological image. The majority of the experts were able to obtain a set of geostatistics which were consistent with the true (measured) statistics of the target image.

Acknowledgements

Firstly, I would like to thank my principle supervisor, Professor Andrew Curtis, for his support, advice and encouragement throughout my PhD. I would also like to thank Dr Mark Chapman for his advice and supervision, particularly during the first year of my PhD. I thank Professor Ian Main for his support and guidance as my PhD advisor. I also thank the administrative staff at the School of GeoSciences for their help during my time in the school.

I would like to acknowledge the Geoscience Research Centre (GRC) at TOTAL E&P UK who sponsored my PhD. I particularly appreciate the support of my external supervisor there, Dr Mohammad Shahraeeni, who was of great help to me throughout my PhD. I would also like to thank Pierre Thore, the head of Geophysics at the GRC, for stimulating discussions about my project and giving me the opportunity to apply some of my ideas during an internship at the GRC. Additionally, I would like to thank all members of the GRC, for their help and guidance throughout that internship.

Finally I thank my family, and especially my fiancée Laura to whom this thesis is dedicated, for supporting me through the duration of the project.

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview

Seismic data can be used to infer the physical properties of the subsurface. This process, herein referred to as *seismic inversion*, is particularly valuable for reservoir characterisation (Haas and Dubrule, 1994). The parameters that are to be inverted for depend on the context of the inversion. Viscoelastic parameters can be directly related to, and hence inferred from, the seismic data using the physics of wave propagation. However, it is also desirable to infer parameters describing geological and petrophysical properties of interest in the subsurface, henceforth referred to as *geological* parameters, since such parameters can be used in reservoir appraisal, development and production processes. They cannot be directly related to the seismic data by wave theory, but may be related to the viscoelastic parameters using theoretical rock-physics or statistical models derived from empirical data.

The seismic data is usually derived from the results of large-scale surface seismic surveys where the seismic wavefield is recorded at the surface. In essence, this raw (henceforth 'pre-stack') seismic data is inverted to estimate the viscoelastic parameters and then subsequently the geological parameters can be estimated by inverting the theoretical/statistical relationship between them and the viscoelastic parameters. However, the physics of wave-propagation in viscoelastic media is complex and direct full-waveform inversion of pre-stack data is costly and unstable, especially when applied to data containing high-frequency information (Virieux and Operto, 2009). Thus the pre-stack data is usually processed first such that it can be related directly to the viscoelastic parameters by the more computationally-tractable

physics of amplitude-versus-angle (AVA) analysis (Tsvankin et al., 2010). This so-called *AVA-type* data then constitutes the seismic data which is inverted for reservoir characterisation. Additionally, AVA-type data is usually processed with the intention of removing the effects of viscoelasticity, thus it is assumed henceforth that only the *elastic parameters* can be inferred from such data.

A well-posed inverse problem is defined as one for which a unique solution exists that varies smoothly with the value of the observed data. Thus the elastic inverse problem is inherently ill-posed, since an infinite number of subsurface elastic models will fit the observed seismic data (Thore, 2013). This is because seismic noise exists at all frequencies and as the seismic wavefield propagates through the subsurface high frequencies are attenuated to a greater extent than low frequencies to the point that noise obliterates signal at very high frequencies (Pendrel, 2001). Seismic sources are generally poor at producing low frequencies and these frequencies tend to be damped by seismometers, thus noise also begins to dominate signal at low frequencies (Barzilai et al., 1998). Furthermore, reflection seismic data is only sensitive to contrasts in the elastic parameters in the subsurface, thus the absolute values of those parameters cannot be determined uniquely (or equivalently, the zero-frequency or mean component of the elastic parameter model cannot be determined) from such data.

The inverse relationship between the elastic parameters and the geological parameters is also generally non-unique. Fundamentally, this is because both the elastic and geological parameters are defined as bulk properties of the subsurface rock (Spikes et al., 2007). This is necessary because the physical structure of the subsurface cannot be determined from the seismic data, or feasibly modelled, at infinitely high resolution. Thus usually the subsurface is modelled as a grid of cells of finite size within which the bulk elastic and geological properties of the rock are to be determined. The choice of the size of the cells is usually made dependent upon the frequency content of the seismic data and the availability of other sources of data such as well data. However, the size of the cells is invariably greater than the smallest scale of heterogeneity in natural subsurface rocks (Mavko et al., 2009). Thus a range of different physical configurations of the rock, including those with differing bulk geological parameters, can give rise to the same bulk elastic parameters within a model cell. For example, a given measurement of the bulk elastic parameters in a cell may correspond to a wide variety of different values for bulk porosity, depending upon the distribution of porosity and the characteristics of the rock matrix within

that cell.

Thus the elastic, geological and overall process of seismic inversion are ill-posed inverse problems. Fortunately, there are always other sources of information about the geological and elastic parameters which can help to constrain their values. This information constitutes so-called prior information, and the combination of this with the seismic data, to form an estimate of the parameters which correctly characterises uncertainty, is achieved using Bayesian methods. The output of such techniques is an estimate of the *posterior* probability distribution, which describes the probability of all of the different possible values for the elastic and geological parameters, given the observed data (Buland and Omre, 2003a) and the prior information. Bayesian seismic inversion is challenging in practice since it requires the choice of forward, prior and data-error models which accurately represent the information available. Furthermore, the computational cost of Bayesian inversion generally scales with the accuracy of these models. Thus this thesis concentrates on developing methods which improve the efficiency and efficacy of current methods for Bayesian seismic inversion for reservoir characterisation.

The rest of this introductory chapter describes the Bayesian seismic inversion problem in general, the problems associated with solving it, and the contribution of this thesis to the field. In section 1.2 the cellular grid and variables, used to model the subsurface in this thesis, are described. In section 1.3 the geological parameters and their relation to the elastic parameters are described. Section 1.4 describes the AVA-type data, and the forward model which relates it to the elastic parameters. Section 1.5 discusses the general framework for solving Bayesian seismic inversion problems, and the difficulties associated with solving it. Sections 1.6 and 1.7 then describe the particular approach to inversion that is the focus of this thesis. A number of research topics are identified throughout this chapter; section 1.8 outlines these topics and how these are addressed in the rest of the thesis. It should be noted that there are many alternative interpretations of the seismic inversion problem depending mainly on the type of supplementary data available (e.g., well data or other geophysical survey results). The discussion presented here, although general, concentrates on the problem of inverting seismic data alone, and thus does not cover all such interpretations.

1.2 The cellular model-grid and variable notation

In this thesis we use cellular grids to model the subsurface. Depending upon application, these may be one (1-D), two (2-D) or three (3-D) dimensional grids. Cells within 3-D grids are described by three coordinates with $x \in [1, 2, ..., X-1, X]$ and $y \in [1, 2, ..., Y-1, Y]$ describing the position of the cell in two lateral directions, and $z \in [1, 2, ..., Z-1, Z]$ describing the position of the cell in the vertical direction. X, Y and Z are the dimensions of the grid. By definition x, y and z are unit-less: they simply represent the number of the cell in their respective direction (the size of cells, and their absolute positions will be specified where necessary for real data). Thus the total number of cells in the grid is $M = Z \times X \times Y$. Indices are used to reference cells within the grid, defined as $i = (Z \times X \times (y-1)) + (Z \times (x-1)) + z$.

Cells within 2-D grids are described by two coordinates with $x \in [1, 2, ..., X-1, X]$ describing lateral position and $z \in [1, 2, ..., Z-1, Z]$ describing vertical position, where X and Z are the dimensions of the grid. The total number of cells in the grid is $M = Z \times X$, and a 2-D grid index is defined as $i = (Z \times (x-1)) + z$. 1-D grids are useful for describing single traces in the subsurface, thus cells in such grids are referenced by a single coordinate $z \in [1, 2, ..., Z-1, Z]$ describing vertical position, or equivalently index, in the grid. In any case, the set of all indices in a grid (1-D, 2-D or 3-D) is written $\mathcal{H} = \{1, 2, ..., M-1, M\}$.

A vector describing the bulk elastic parameters $\mathbf{e}_i$ is assigned to each cell i in a grid. In general $\mathbf{e}_i = [I_P, I_S, \rho]_i$ where I_S is S-wave impedance, I_P is P-wave impedance and ρ is density. We will use the notation $\mathbf{e} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_M]$ to refer to a vector containing all elastic parameter vectors in a grid (where the subscripts refer to the index of a cell in the grid). It will be useful later also to refer to the elastic parameters down the z dimension at a given lateral position $\mathbf{x} = [x, y]$ (for a 3-D grid) using the notation $\mathbf{e}_\mathbf{x} = \{ \mathbf{e}_{x,y,z} \mid z \in [1, 2, ..., Z-1, Z] \}$, where $\mathbf{e}_{x,y,z}$ is the elastic parameter vector at the cell with coordinates $[x, y, z]$.

The geological parameters used for reservoir characterisation can be discrete or continuous, or a combination of the two. Discrete geological parameters usually describe a single categorical variable for a cell such as facies or rock-type. Thus to model these we assign a discrete variable g_i to each cell in the grid, and in general we will we use the notation $\mathbf{g} = [g_1, g_2, ..., g_M]$ to refer to a vector containing all such parameters in the grid (where the subscripts refer to the index of a cell in the grid). Continuous geological parameters usually describe multiple bulk physical

properties of the rock such as porosity or water saturation. Thus to model these we assign a vector of L continuous geological parameters $\mathbf{m}_i = [m_1, m_2, ..., m_L]_i$ to each cell in the grid. Generally, the set of all such vectors in the grid is written $\mathbf{m} = [\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_M]$ (where, again, the subscripts refer to the index of a cell in the grid). The distinction between continuous and discrete geological parameters will be useful later, but for convenience in the rest of this introductory chapter g_i is used to represent geological parameters *in general*. All equations written using g_i are equally valid for $\mathbf{m}_i$, or the combination $[g_i, \mathbf{m}_i]$ (with substitution of appropriate integration and summation limits). The g_i variables associated with all cells with indices in a set $\mathcal{S}$ are referenced using the notation $\mathbf{g}_\mathcal{S} = \{g_i \mid i \in \mathcal{S}\}$, where $\mathcal{S} = [1, 4, 6]$, for example. This notation is used in the same way for $\mathbf{e}_i$.

Both g_i and $\mathbf{e}_i$ are interpreted as random variables, and we write their sample spaces as $\mathcal{G}$ and $\mathbb{R}^3$, respectively. We assume that the variables have identical sample spaces in each cell, thus the sample spaces of $\mathbf{g}$ and $\mathbf{e}$ may be written $\mathcal{G}^M$ and $\mathbb{R}^{3M}$ respectively, where the M exponent implies that the sample space for a single cell is taken to the power of the number of cells in the grid. For the special case where g_i is discrete, the size of $\mathcal{G}$ can be written $|\mathcal{G}|$, and the size of $\mathcal{G}^M$ can be calculated as

$$|\mathcal{G}^M| = |\mathcal{G}|^M. \tag{1.1}$$

Probability mass and density functions may be defined over discrete (e.g., $\mathbf{g}$) and continuous (e.g., $\mathbf{e}$) variables, respectively. A so-called mixed probability distribution may also be defined over a combination (e.g., $[\mathbf{g}, \mathbf{e}]$) of these two variable types. We use the notation $p()$ to denote each of these types of probability distribution interchangeably. In this thesis, we will frequently refer to *parametric* distributions, which are a probability mass, density or mixed functions of closed-form which may be evaluated analytically, and whose normalisation constant may also be calculated analytically.

The above notation will be used throughout this introductory chapter. However, some slight modifications must be made to this notation within Chapters 2-5. Thus for the avoidance of doubt, each of these chapters contains a section which describes the notation used therein.

1.3 The geological parameters

The $\mathbf{e}_i$ variable can be related to g_i using a forward model. This can be established using empirical measurements from lab- or well- data to constrain a purely statistical model (Chang et al., 2006), or rock-physics theory can be used to establish a deterministic relationship (Avseth et al., 2005). Rock-physics models are constructed to predict $\mathbf{e}_i$, under some assumptions about the micro-structure of the rock, given g_i (Mavko et al., 2009). For example, a model may be designed to predict $\mathbf{e}_i$ for a rock which has a consolidated, homogeneous matrix with round pores, given its bulk porosity. Given the heterogeneity of natural rocks, it is rare that such models are an accurate depiction of reality and thus they can suffer from epistemic errors. Thus, in general, the forward relationship established with either method is uncertain. Fundamentally this is due to the definition of these as *bulk* parameters: g_i does not describe the exact physical structure of the rock within cell i and therefore cannot exactly predict $\mathbf{e}_i$. Given this uncertainty, it is appropriate to use a conditional probability distribution $p(\mathbf{e}_i|g_i)$ (henceforth, the *cell-wise geological likelihood*), to describe the forward relationship at each cell. We henceforth assume that a single such distribution is applicable throughout the model grid (i.e., the distribution is invariant to i).

It is often assumed (Mukerji et al., 2001) that the elastic parameters in a given cell are completely explained by the geological parameters in that cell, thus the specification of *any* other variable in the grid yields no more useful information about the elastic parameters at that cell. This is referred to as the *local geological likelihood* property henceforth. Mathematically, it allows us to write

$$p(\mathbf{e}_i|g_i, \mathbf{g}_{\subseteq\mathcal{H}\setminus i}, \mathbf{e}_{\subseteq\mathcal{H}\setminus i}) = p(\mathbf{e}_i|g_i) \tag{1.2}$$

where the notation $\subseteq \mathcal{H}\setminus i$ should be read as 'any set of indices in the grid which does not include i' (thus $\mathbf{g}_{\subseteq\mathcal{H}\setminus i}$ is the set of all g_i variables in those cells).

The property is true if we have derived a causal forward relationship between $\mathbf{e}_i$ and g_i, which is usually the case and which we assume to be the case henceforth. The property permits simplification of the so-called *joint geological likelihood* distribution $p(\mathbf{e}|\mathbf{g})$, which describes the joint conditional probability of $\mathbf{e}_i \; \forall \; i$ given $g_i \; \forall \; i$ (where henceforth $\forall \; i$ is used as the abbreviation of $\forall \; i \; \in \; \mathcal{H}$). Specifically, we are now able to write it as a product of each of the individual cell-wise geological likelihoods,

which is shown by decomposing $p(\mathbf{e}|\mathbf{g})$ using elementary probability identities as

$$p(\mathbf{e}|\mathbf{g}) = \prod_{i=1}^{M} p(\mathbf{e}_i|\mathbf{g}, \mathbf{e}_{<i}) = \prod_{i=1}^{M} p(\mathbf{e}_i|g_i). \tag{1.3}$$

where the subscript $< i$ indicates the set of all indices in $\mathcal{H}$ less than i (thus $\mathbf{e}_{<i}$ is the set of all elastic parameter vectors in cells with index less than i), and the second inequality holds because of the assumption of the local geological likelihood property.

1.4 The AVA-type data

The AVA-type data is formed by first processing pre-stack seismic data such that it contains only primary P-wave reflection events. Pre-stack migration is then applied such that common mid-point gathers can be assumed to represent the response of a locally 1-D earth (Castagna, 1993). This data is then converted from the offset- to incident angle- domain with respect to the normal to the discontinuity surface which generated the reflection event. Each reflection event should be normalised as if the incident wave had constant amplitude irrespective of the position of the discontinuity surface, which requires compensation for the effects of intrinsic (viscoelastic) and extrinsic (scattering and spreading) attenuation (Hampson, 1991). Ideal AVA data then comprises traces, $\mathfrak{d}_{\theta,\mathbf{x}}$ describing amplitudes of the reflected energy at all vertical positions z and incidence angle θ, at a given lateral position $\mathbf{x} = [x, y]$. Since it is assumed to represent the response of a locally 1-D earth, it follows that we can assume that a single trace of data is dependent only on the elastic parameter profile with z (depth) at the same lateral position, thus we write $\mathfrak{d}_{\theta,\mathbf{x}}(\mathbf{e}_\mathbf{x})$. Approximations to such data are now a standard output of seismic processing (Virieux and Operto, 2009), but significant error will exist in such data due to inaccuracy in the amplitude compensation (Hubral, 1983) and angle-to-offset transformation (Sava and Fomel, 2003). As noted above, the compensation for the effects of viscoelasticity means that information regarding the viscoelastic properties of the subsurface is ignored.

We may model the data at a single lateral position $\mathfrak{d}_{\theta,\mathbf{x}}(\mathbf{e}_\mathbf{x})$ by convolving an angle-dependent wavelet with a reflectivity series (Hampson et al., 2005). The reflectivity series can be calculated using approximations of the Zoeppritz equations (Shuey, 1985). For example, the reflectivity (at incidence angle θ) can be calcu-

lated using the weak contrast approximation to the reflection coefficient (Aki and Richards, 2002) as

$$\begin{aligned} r_\theta =& \frac{1}{2}\left(\frac{\Delta\alpha}{\bar{\alpha}} + \frac{\Delta\rho}{\bar{\rho}}\right) \\ &+ \left(\frac{1}{2}\frac{\Delta\alpha}{\bar{\alpha}} - 4\frac{\bar{\beta}^2}{\bar{\alpha}^2}\frac{\Delta\beta}{\bar{\beta}} - 2\frac{\bar{\beta}^2}{\bar{\alpha}^2}\frac{\Delta\rho}{\bar{\rho}}\right)\sin^2\theta \\ &+ \frac{1}{2}\frac{\Delta\alpha}{\bar{\alpha}}\left(\tan^2\theta - \sin^2\theta\right) + \epsilon_\theta \end{aligned} \tag{1.4}$$

where $\alpha = \frac{I_P}{\rho}$ is the P-wave velocity, $\beta = \frac{I_S}{\rho}$ is the S-wave velocity and ϵ_θ represents an error term comprising missing higher order terms in the Taylor expansion used in this approximation. The overbar and delta symbols denote averages and differences of these quantities over the discontinuity, respectively. Since we assume that the data are generated from a locally 1-D earth we model them using a reflectivity series calculated in the vertical z direction. We calculate the reflectivity at each vertical position z with respect to $z-1$, yielding a vertical reflectivity series vector $\mathbf{r}_{\theta,\mathbf{x}}(\mathbf{e_x})$ for angle of incidence θ. Given this reflectivity vector, the data is modelled using the convolution $\mathfrak{d}_{\theta,\mathbf{x}}(\mathbf{e_x}) = \mathbf{r}_{\theta,\mathbf{x}}(\mathbf{e_x}) * \mathbf{w}_\theta$, where $\mathbf{w}_\theta$ is a vector specifying the appropriate wavelet for angle of incidence θ. This convolution can be written as a matrix multiplication (Buland and Omre, 2003b)

$$\mathfrak{d}_{\theta,\mathbf{x}}(\mathbf{e_x}) = \mathbf{s}_\theta \mathbf{r}_{\theta,\mathbf{x}}(\mathbf{e_x}) \tag{1.5}$$

where $\mathbf{s}_\theta$ is the Toeplitz matrix for the wavelet vector $\mathbf{w}_\theta$ (padded with an appropriate number of zeroes to ensure that the matrix multiplication represents convolution with the reflectivity series). Variation in the wavelet between angles of incidence is assumed to arise from variation in dispersion caused by differing ray path length and trajectory (Buland and Omre, 2003c). It is usually assumed that a single, constant-in-time wavelet $\mathbf{w}_\theta$ is appropriate for each angle of incidence (or even a range of angles). This is usually acceptable if the vertical extent of the region of interest for which we invert (a reservoir interval, for instance) is small and hence little dispersion may occur within that interval. In practice so-called angle-stacks are constructed where the migrated seismic data is stacked over angular ranges of incidence, rather than data vectors which are valid for single angles of incidence. Such stacks are easier to generate from pre-stack data and increase the signal-to-noise ratio. In effect they

are formed by stacking $\mathfrak{d}_{\theta,\mathbf{x}}(\mathbf{e_x})$ vectors for ranges of θ. Henceforth, we use angular ranges called 'near' ($\theta = 6 - 16°$), 'mid' ($\theta = 16 - 26°$) and 'far' ($\theta = 26 - 36°$), and notation such as $\mathfrak{d}_{mid,\mathbf{x}}(\mathbf{e_x})$ to mean the angle-stack data for the mid-range angles.

It is convenient to be able to write all the data down a trace, and its relation to the elastic parameters, as a single matrix equation. This can be derived by analogy to equation 1.5, as

$$\mathbf{f}(\mathbf{e_x}) = \mathbf{d_x}(\mathbf{e_x}) = \mathbf{SR}(\mathbf{e_x}) + \mathbf{n} \tag{1.6}$$

where $\mathbf{S}$ is a block-matrix formed by concatenating the wavelet Toeplitz matrices for each angular range, $\mathbf{R}(\mathbf{e_x})$ is a single reflectivity vector constructed by concatenating the reflectivity vectors for each angular range, and $\mathbf{d_x}(\mathbf{e_x})$ is the AVA-type data for each angular range, arranged into a single vector. These vectors and matrices are defined in Appendix A. In equation 1.6 the notation $\mathbf{f}(\mathbf{e_x})$ indicates that this equation represents the forward physics of the problem. $\mathbf{n}$ is a vector (with dimension equal to $\mathbf{SR}(\mathbf{e_x})$) of zero-mean Gaussian noise distributed as

$$\mathbf{n} \sim \phi(\mathbf{0}, \mathbf{\Sigma_d}) \tag{1.7}$$

where $\phi(\mathbf{0}, \mathbf{\Sigma_d})$ is a multivariate Gaussian distribution with mean vector $\mathbf{0}$ and covariance matrix $\mathbf{\Sigma_d}$, which is the error covariance matrix describing the random error on the data for all angle stacks. Both $\mathbf{\Sigma_d}$ and $\mathbf{S}$ can be estimated for the seismic data using well-tying techniques (see e.g., Bo et al., 2013), and we assume henceforth that these parameters are estimated separately from the seismic inversion procedure. For simplicity we also assume henceforth that both $\mathbf{\Sigma_d}$ and $\mathbf{S}$ are constant with respect to lateral position $\mathbf{x}$. However, both quantities may in fact vary across the extent of the seismic survey; indeed they are often treated as random variables within seismic inversion (Buland and Omre, 2003c). Processing errors (e.g., in the offset to angle transformation) in the AVA-type data cannot be estimated independently, and will contribute to the seismic noise estimated in the well-tying procedure (i.e., $\mathbf{\Sigma_d}$).

It is appropriate to write the uncertain AVA forward relation at a single lateral position $\mathbf{x}$ using the conditional probability distribution $p(\mathbf{d_x}|\mathbf{e_x})$, which is the probability of observing the AVA-data $\mathbf{d_x}$ given the elastic parameter configuration $\mathbf{e_x}$. Given that the seismic noise $\mathbf{\Sigma_d}$ is assumed to be distributed normally, then this

distribution can be written as a Gaussian function thus

$$p(\mathbf{d_x}|\mathbf{e_x}) = (2\pi)^{-\frac{k}{2}}|\mathbf{\Sigma_d}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{d_x} - \mathbf{f}(\mathbf{e_x}))^T \mathbf{\Sigma_d}^{-1} (\mathbf{d_x} - \mathbf{f}(\mathbf{e_x}))\right) \tag{1.8}$$

where $\mathbf{f}(\mathbf{e_x})$ is the AVA forward function (equation 1.8) and k is the dimensionality of the data vector. We henceforth assume that the errors on the AVA-data are approximately independent with respect to lateral position $\mathbf{x}$. Thus we may write the joint probability of all of the AVA-type data in the grid $\mathbf{d}$, as the product

$$p(\mathbf{d}|\mathbf{e}) = \prod_{\forall\, \mathbf{x}} p(\mathbf{d_x}|\mathbf{e_x}) \tag{1.9}$$

where $\forall\ \mathbf{x}$ implies the set of all lateral positions $[x, y]$ (or x in the 2-D case) in the grid. The distribution in equation 1.9 is referred to as the *elastic likelihood* distribution henceforth.

1.5 The general Bayesian framework for seismic inversion

1.5.1 The posterior

In this section we discuss how $\mathbf{d}$ may be inverted for $\mathbf{g}$ and $\mathbf{e}$ in a Bayesian framework. Ideally, we aim to determine the so-called *joint* posterior probability distribution $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$ which is the joint probability of the elastic $\mathbf{e}$ and geological $\mathbf{g}$ parameters, given the AVA-type data $\mathbf{d}$ (Bosch et al., 2010). It can be expressed using Bayes' rule (see e.g., Ulrych et al., 2001) as

$$p(\mathbf{e}, \mathbf{g}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{e}, \mathbf{g})p(\mathbf{e}, \mathbf{g})}{p(\mathbf{d})}, \tag{1.10}$$

where $p(\mathbf{e}, \mathbf{g})$ is the joint prior probability distribution, which describes the information known about $\mathbf{e}$ and $\mathbf{g}$ independently of the data. The $p(\mathbf{d})$ term on the right hand side of equation 1.10 is a constant since it is a function only of the data $\mathbf{d}$, which is observed and hence fixed in this inversion context. It is referred to as the normalising constant since it may be shown that $p(\mathbf{d}) = \int_{\mathbb{R}^{3M}} \sum_{\mathbf{g}\in\mathcal{G}^M} p(\mathbf{d}|\mathbf{e}, \mathbf{g})p(\mathbf{e}, \mathbf{g})d\mathbf{e}$ (Sambridge et al., 2006).

It is assumed that $\mathbf{d}$ can be completely explained by $\mathbf{e}$ by the physics of elastic wave propagation. Thus the final distribution in equation 1.10 can be written using conditional independence as

$$p(\mathbf{d}|\mathbf{e},\mathbf{g}) = p(\mathbf{d}|\mathbf{e}), \tag{1.11}$$

since the specification of $\mathbf{g}$ provides no additional information about $\mathbf{d}$ (to that provided by $\mathbf{e}$). Thus the likelihood in equation 1.10 is equivalent to the elastic likelihood given in equation 1.9. All distributions in the joint posterior in equation 1.10 have now been defined apart from the joint prior distribution $p(\mathbf{e},\mathbf{g})$, which is discussed in the next section.

1.5.2 The prior

Information always exists about the geology of the subsurface independently of the seismic data, which can be used to inform the inversion (Curtis and Wood, 2004). This information is often specific to the region of interest, such as the expected spatial distribution of facies in the subsurface (Kolbjørnsen et al., 2013). However, even if such specific information is absent then there is at least information in the sense that the general laws and concepts of geology can be applied, such as those describing the geometry of sedimentological or structural features (Torres-Verdin et al., 1999) or how those features are created (Hill et al., 2009). Such prior information about $\mathbf{g}$ can be codified within the probability distribution $p(\mathbf{g})$ (henceforth the *geological prior*). This geological information can in turn be transformed into information about the elastic parameters using $p(\mathbf{e}|\mathbf{g})$ (equation 1.3); the joint prior distribution required by equation 1.10 can then be constructed using the probability identity

$$p(\mathbf{e},\mathbf{g}) = p(\mathbf{e}|\mathbf{g})p(\mathbf{g}). \tag{1.12}$$

Usually, $p(\mathbf{g})$ is defined using geostatistical methods. In two-point geostatistics the variogram is used to specify the variance of the difference between values of g_i at *two* different positions in the grid, as a function of the relative position of the two points. Such a function can be determined empirically for a so-called training image of $\mathbf{g}$, which is an image of $\mathbf{g}$ designed to demonstrate all of the geological features which are expected of the geological parameters, given the available prior information. The empirical variogram can then be used to define $p(\mathbf{g})$ using either a

non-parametric or parametric approach. In the former case $p(\mathbf{g})$ is defined as being equiprobable for all realisations of $\mathbf{g}$ which are consistent (within some tolerance) with the empirical variogram measured (Olea, 1999, p. 154). In the parametric approach, it is assumed a-priori that $p(\mathbf{g})$ may be written as some parametric function. For example, one may assume that $p(\mathbf{g})$ is a Gaussian distribution for which the mean vector may be estimated directly from the training image, and the covariance matrix can be calculated from the empirical variogram calculated for the training image (Olea, 1999, p. 146).

The use of two-point geostatistics (i.e., the variogram) is not a natural choice for describing the variation of discrete, particularly categorical, geological parameters (Caers, 2005, p. 24) (that is to say, the above methods are most useful when we are dealing with continuous geological variables, $\mathbf{m}$). Furthermore, it cannot encapsulate higher-order statistical information about $\mathbf{g}$ (Remy et al., 2009, p. 50). Multi-point geostatistics is designed to capture such sophisticated information, and is more amenable to the modelling of discrete geological variables. In practice, such multi-point statistical information is specified using probability distributions. For example, a probability distribution can be defined which describes the probability of the geological parameters at a single cell, conditioned upon the value of the parameters in the surrounding cells (Remy et al., 2009, p. 64), which is written

$$p(g_i|\mathbf{g}_{\mathcal{H}\backslash i}) = \frac{p(\mathbf{g})}{p(\mathbf{g}_{\mathcal{H}\backslash i})}, \tag{1.13}$$

where $\mathcal{H}\backslash i$ is the set of all indices in the grid except i. This distribution is then considered to be stationary with respect to position i in the subsurface grid. Such distributions, henceforth referred to as *full conditionals* (Besag, 1974), can be determined from a training image. Full conditionals, and their relationship to $p(\mathbf{g})$, are discussed in detail later (section 4.4), but for now we assume that the specification of the stationary distribution $p(g_i|\mathbf{g}_{\mathcal{H}\backslash i})$ permits evaluation of a corresponding prior probability distribution $p(\mathbf{g})$.

The above discussion assumes that appropriate training images are available for the extraction of statistics with which we may define $p(\mathbf{g})$. Photographs (Dueholm and Olsen, 1993) or even geophysical survey results (Caers et al., 1999) of analogue formations can be used to construct training images directly (Pringle et al., 2004), but their relevance depends on the similarity of the analogue and target formation

geology (Ringrose et al., 1999). It is widely accepted that a lack of suitable analogue formation data is a significant problem (Cui et al., 1995; Kerry and Oliver, 2007; Truong et al., 2013). Thus alternatively, training images may be constructed using process- or object- based models with the input of a geological expert. However, it can be a costly task, in terms of computation and expert time, to generate a training image that sufficiently well illustrates the experts' knowledge of $\mathbf{g}$. Thus one of the objectives of this thesis is to develop a method for efficiently eliciting multi-point geostatistical information directly from a geological expert, without the need for this costly intermediary step.

It is important to note that $p(\mathbf{e}, \mathbf{g})$ will not in general be of parametrised form. This is true even if $p(\mathbf{g})$ is defined parametrically since the multiplication in equation 1.12 will in general not yield a parametrised form. Additionally, it is likely that $p(\mathbf{e}, \mathbf{g})$ is multi-modal in form if $p(\mathbf{g})$ is defined using multi-point geostatistics or non-parametric two-point methods, since there is not necessarily any connection between euclidean distance and geological similarity within $\mathbf{g} \in \mathcal{G}^M$ (Pham, 2010).

1.5.3 Fundamental problems in determining the posterior

For 2-D or 3-D grids, the number of cells M will often be large, thus the dimensionality of the sample spaces of $\mathbf{g}$ and $\mathbf{e}$ are usually very large. Fundamentally this means that computations on these parameter spaces are very intensive (in terms of the required memory and number of calculations) even for the simplest of geological parameters. For example, consider a discrete geological parameter at each cell describing rock type $g_i \in \mathcal{G} = [\text{reservoir}, \text{non-reservoir}]$. This implies that $|\mathcal{G}| = 2$. However even for small models $M > 10^3$, thus using equation 1.1 we have that $|\mathcal{G}^M| = |\mathcal{G}|^M > 10^{301}$, and more typical industrial scale models have $M \sim 10^6 - 10^9$.

Furthermore, from the preceding discussions it is clear that neither $p(\mathbf{d}|\mathbf{e})$ nor $p(\mathbf{e}, \mathbf{g})$ are likely to be parameterised distributions. Thus in general $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$ (equation 1.10) cannot be determined parametrically (George et al., 1993), and additionally the normalising constant $p(\mathbf{d})$ cannot be calculated analytically. However, both $p(\mathbf{d}|\mathbf{e})$ and $p(\mathbf{e}, \mathbf{g})$ may be evaluated (up to a constant of proportionality) for a given realisation of $\mathbf{e}$ and $\mathbf{g}$, using equations 1.9 and 1.12, respectively. Thus to characterise $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$ it might be possible to discretise the entire joint parameter space $\mathcal{G}^M \times \mathbb{R}^{3M}$ and systematically evaluate and store the value of the numerator of equation 1.10 throughout this discretisation (and the values retained could then also be

used to perform numerical integration to obtain $p(\mathbf{d})$). However, the size of the joint parameter space $\mathcal{G}^M \times \mathbb{R}^{3M}$ is usually very large. Thus such an operation would be extremely inefficient because it requires exploration of the entire extent of these large parameter spaces.

There are then two practical approaches for characterising $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$, the first of which may be labelled *deterministic inversion*. Methods of this class seek to obtain the maximum-a-posteriori (MAP) estimate, which is the realisation of $\mathbf{g}$ and $\mathbf{e}$ with maximum posterior probability (Bosch et al., 2012). The MAP can be found by gradient-ascent methods using the gradient vector (usually calculated numerically) of the numerator in equation 1.10. Uncertainty can then be evaluated by estimating the local posterior variance about the MAP estimate (Gubbins, 2004). The second class of inversion methods may be labelled *stochastic* since they seek to obtain a set of representative realisations (samples) from the posterior (Srivastava and Sen, 2010). For stochastic inversions Monte-Carlo (MC) methods are appropriate since they permit random sampling from the $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$. Such sampling algorithms require only that the numerator of the posterior can be evaluated up to a constant of proportionality (Mosegaard and Sambridge, 2002).

It is clear that the cost of an iteration of a stochastic or deterministic method is proportional to the cost of evaluating the numerator of equation 1.10, that is evaluating both $p(\mathbf{d}|\mathbf{e})$ and $p(\mathbf{e}, \mathbf{g})$. In general both of these distributions can be costly to evaluate. Additionally, both methods are susceptible to local convergence problems, thus in general the more multi-modal the posterior the greater the number of iterations/samples required to obtain a good solution in both cases (Grana et al., 2011). Thus, roughly-speaking, it can also be said that the cost of inversion scales with any multi-modality induced in the posterior by the prior and likelihood.

In general, it is desirable to use multi-point geostatistics to specify $p(\mathbf{g})$, since it can represent the available prior information most accurately. However, in this case it is likely that $p(\mathbf{e}, \mathbf{g})$ will be multi-modal, which will in turn induce multi-modality in $p(\mathbf{e}, \mathbf{g}|\mathbf{d})$. Thus this means that the direct estimation of the joint posterior by deterministic or stochastic inversion methods can be expensive. However, there is a practical method that reduces the computational cost of inversion yet permits multi-point geostatistics to be applied. This so-called 'two-stage' inversion, explicitly splits the problem of posterior estimation into *elastic inversion* and *geological inversion* stages (Bosch et al., 2010). In solving the former inverse problem, generally a simplified prior distribution is employed which promotes efficient and stable

inversion (Filippova et al., 2011). However, in solving the latter problem, sophisticated multi-point geostatistics can be used to specify the geological prior. The methodologies developed for Bayesian seismic inversion in this thesis are made in the context of this method, thus it is described in greater detail in the next section.

Nevertheless, a number of methods do exist for directly estimating the joint posterior in equation 1.10. González et al. (2007) used a stochastic method to sample from the joint posterior, where the geological prior was specified using multi-point geostatistics. However, this prior was not defined in a probabilistic way, and the stochastic algorithm itself was dependent upon the availability of well data. Another example of such a 'single-stage' algorithm is that of Rimstad et al. (2012), who used full conditionals to specify a multi-point geostatistical prior, but the conditional dependence within the full conditional distribution (equation 1.13) was limited to only a small set of neighbouring cells. The method also assumed a linearisation of $\mathbf{f}(\mathbf{e})$, since it can be shown that equation 1.4 is only weakly non-linear in $\mathbf{e}$. The probability perturbation method of Caers and Hoffman (2006) is a general method developed for single-stage inversion, but relies upon some quite restrictive assumptions about independence between parameters in the model (the so-called 'tau-model') for its derivation.

1.6 Two-stage Bayesian seismic inversion

The joint posterior in equation 1.10 can be split explicitly into elastic and geological inversion parts (Bosch et al., 2010) by rewriting it using elementary probability identities as

$$p(\mathbf{e}, \mathbf{g}|\mathbf{d}) = p(\mathbf{e}|\mathbf{d})p(\mathbf{g}|\mathbf{e}, \mathbf{d}) = p(\mathbf{e}|\mathbf{d})p(\mathbf{g}|\mathbf{e}) \tag{1.14}$$

where the second equality holds since $\mathbf{g}$ can only affect $\mathbf{d}$ via changes in $\mathbf{e}$, thus once $\mathbf{e}$ is specified, $\mathbf{d}$ is redundant in $p(\mathbf{g}|\mathbf{e}, \mathbf{d})$. Equation 1.14 separates the joint posterior into an elastic posterior $p(\mathbf{e}|\mathbf{d})$ and a geological posterior $p(\mathbf{g}|\mathbf{e})$. The former may be written using Bayes' rule as

$$p(\mathbf{e}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{e})p(\mathbf{e})}{p(\mathbf{d})}, \tag{1.15}$$

where $p(\mathbf{d}|\mathbf{e})$ is the elastic likelihood, $p(\mathbf{e})$ is the so-called elastic prior distribution and $p(\mathbf{d}) = \int_{\mathbb{R}^{3M}} p(\mathbf{d}|\mathbf{e})p(\mathbf{e})d\mathbf{e}$ is the normalising constant.

As with the joint posterior in equation 1.10, in general there is no way to determine a parametrised elastic posterior because neither $p(\mathbf{d}|\mathbf{e})$ nor $p(\mathbf{e})$ are parametrised distributions, and the dimensionality of the sample space of $\mathbf{e}$ (i.e., $3M$) prohibits a systematic exploration of the parameter space. Thus the elastic posterior can usually only be characterised using deterministic or stochastic inversion techniques, which scale in cost with the cost of evaluating $p(\mathbf{d}|\mathbf{e})$ and $p(\mathbf{e})$, and the degree of multi-modality induced in $p(\mathbf{e}|\mathbf{d})$ by these distributions.

Thus to reduce the computational cost of (deterministic or stochastic) elastic inversion, a form for $p(\mathbf{e})$ is chosen which has simple structure (i.e., is not multi-modal) and is computationally cheap to evaluate (Dubrule et al., 1998; Lamy et al., 1999). For example, it may be assumed to be a Gaussian distribution (Buland and Omre, 2003b). Such a simplification can be justified since it can be argued that, a-priori, little is known *directly* about the distribution of elastic parameters in the subsurface (except perhaps some bounds on their values and requirements for lateral continuity). However, the information encapsulated by such a simple $p(\mathbf{e})$ may be inconsistent with the geological prior information encapsulated by the geological prior $p(\mathbf{g})$; ideally the elastic prior $p(\mathbf{e})$ would be obtained from the joint prior $p(\mathbf{e}, \mathbf{g})$ by the marginalisation

$$p(\mathbf{e}) = \sum_{\mathbf{g} \in \mathcal{G}^M} p(\mathbf{e}, \mathbf{g}) = \sum_{\mathbf{g} \in \mathcal{G}^M} p(\mathbf{e}|\mathbf{g})p(\mathbf{g}) \tag{1.16}$$

where the second equality is obtained by substitution of equation 1.12. Thus it is clear that an arbitrary choice of $p(\mathbf{e})$ is not necessarily consistent with predefined $p(\mathbf{g})$ and $p(\mathbf{e}|\mathbf{g})$ distributions. For example, it is unlikely that a Gaussian elastic prior $p(\mathbf{e})$ would ever arise naturally if $p(\mathbf{g})$ were defined using multi-point geostatistics (e.g., equation 1.13).

Thus it is clear that the choice of such a simple elastic prior, whilst promoting efficiency, can represent a significant loss of prior information about the elastic parameters. Importantly, since determination of the geological posterior $p(\mathbf{g}|\mathbf{e})$ is dependent upon the results of elastic inversion, this can also effect any inferences made about the geological parameters, regardless of the accuracy of the geological prior supplied for geological inversion. Thus in this thesis develop an efficient method

that transforms the results of Bayesian elastic inversion obtained using only a simple elastic prior $p(\mathbf{e})$ (defined as a Gaussian), to new estimates of $\mathbf{e}$ which incorporate complex (multi-point geostatistical) prior information. However, the method which we choose to apply is fundamentally computationally expensive, thus we must apply approximations to it which reduce the accuracy of the final posterior estimate obtainable by the method.

In two-stage seismic inversion it is usual for the elastic posterior to be determined using deterministic methods (Francis, 2006), thus a single MAP estimate of $\mathbf{e}$ is determined in the elastic inversion stage. The MAP estimate, denoted $\hat{\mathbf{e}}$, is then used to condition the geological posterior $p(\mathbf{g}|\hat{\mathbf{e}})$ for geological inversion. The geological posterior may then be written using Bayes' rule as

$$p(\mathbf{g}|\hat{\mathbf{e}}) = \frac{p(\hat{\mathbf{e}}|\mathbf{g})p(\mathbf{g})}{p(\hat{\mathbf{e}})} \tag{1.17}$$

where $p(\hat{\mathbf{e}}|\mathbf{g})$ is the joint geological likelihood (equation 1.3), $p(\mathbf{g})$ is the geological prior distribution and $p(\hat{\mathbf{e}}) = \sum_{\mathbf{g}\in\mathcal{G}^M} p(\hat{\mathbf{e}}|\mathbf{g})p(\mathbf{g})$ is the normalising constant. Once again, there is in general no way to determine a parametrised geological posterior because neither $p(\hat{\mathbf{e}}|\mathbf{g})$ nor $p(\mathbf{g})$ are parametrised distributions, and the size of the sample space of $\mathbf{g}$ ($|\mathcal{G}|^M$) prohibits a systematic exploration of the parameter space. Thus again deterministic or stochastic methods must be used to characterise the geological posterior distribution $p(\mathbf{g}|\hat{\mathbf{e}})$, whose cost scales with the cost of evaluating $p(\hat{\mathbf{e}}|\mathbf{g})$ and $p(\mathbf{g})$.

As explained above, unlike for elastic inversion, the prior $p(\mathbf{g})$ used in geological inversion is usually defined using multi-point geostatistics (Bosch et al., 2010), and is thus likely to be multi-modal in form and expensive to evaluate. However, the cost (of a single iteration) of geological inversion is significantly reduced in comparison to elastic inversion, since the likelihood $p(\hat{\mathbf{e}}|\mathbf{g})$ in this case is much cheaper to evaluate than $p(\mathbf{d}|\hat{\mathbf{e}})$ (which requires the costly evaluation of the AVA forward physics, $\mathbf{f}(\hat{\mathbf{e}})$).

Characterising uncertainty in the geological parameters is a key aim of seismic inversion for reservoir characterisation, thus it is usually desirable for a set of $\mathbf{g}$ samples from $p(\mathbf{g}|\hat{\mathbf{e}})$ to be determined (Zhang et al., 2012). Therefore stochastic methods are typically used for the geological inversion stage. However, the MC sampling algorithms used for stochastic inversion generally use a correlated sampling approach. Therefore it is possible that local convergence of the sampler may occur, and hence any estimate of the geological posterior made using the resulting set of

samples may be biased (Belisle, 1998). Thus another objective of this thesis is to investigate this problem further and to develop an alternative sampling algorithm which avoids such bias problems.

The two-stage inversion procedure described in this section is useful since it separates the elastic and geological inversion problems, and enforces an intuitively acceptable simplification of the elastic prior $p(\mathbf{e})$ to reduce the computational cost of elastic (and hence overall seismic) inversion, yet retaining the ability to apply multi-point geostatistical prior information about the geological parameters. However, it does not offer any way to reduce the computational cost of characterising $p(\mathbf{g}|\hat{\mathbf{e}})$. Recently, some effort has been made to do this using so-called analytical Bayesian inversion, which is to say inversion which returns an estimate of the posterior distribution (or some closely related probability distribution) which does not require stochastic methods. Usually these techniques utilize neural networks to perform geological inversion without the need for iterative sampling methods. Such a method is described in the next section.

1.7 Analytical Bayesian geological inversion using neural networks

A so-called neural network can be used to emulate the mapping $\hat{\mathbf{e}}_i \rightarrow p(g_i|\hat{\mathbf{e}}_i)$, which can be used to determine the so-called cell-wise geological posterior distribution $p(g_i|\hat{\mathbf{e}}_i)$ at each cell i in the grid given the elastic parameter estimates $\hat{\mathbf{e}}_i$ at that cell. A neural network can be viewed as a flexible model, mapping a set of inputs to a set of outputs (Roth and Tarantola, 1994). Values for a neural network's adaptable parameters can be found at relatively high computational expense, by a process referred to as *training* (Johansson et al., 1991). Training uses a set of example pairs of the $[g_i, \hat{\mathbf{e}}_i]$ parameters drawn from the joint distribution $p(g_i, \hat{\mathbf{e}}_i)$, to determine values for the network's parameters which cause it to emulate the mapping $\hat{\mathbf{e}}_i \rightarrow p(g_i|\hat{\mathbf{e}}_i)$. The example $[g_i, \hat{\mathbf{e}}_i]$ pairs are obtained by first sampling $g_i \sim p(g_i)$ (the prior geological distribution for a *single* cell in the grid), and then sampling $\hat{\mathbf{e}}_i \sim p(\hat{\mathbf{e}}_i|g_i)$ (as described in section 1.3). Once trained, the neural network can then determine the $p(g_i|\hat{\mathbf{e}}_i)$ distribution corresponding to any $\hat{\mathbf{e}}_i$ vector extremely rapidly and efficiently. Thus this method has been used to efficiently determine $p(g_i|\hat{\mathbf{e}}_i)\ \forall\ i$ for large 3-D grids where deterministic elastic inversion has been used to determine

$\hat{\mathbf{e}}_i \; \forall \; i$ (Shahraeeni et al., 2012).

However, the set of $p(g_i|\hat{\mathbf{e}}_i) \; \forall \; i$ is not a *general* solution to the geological inverse problem (equation 1.17). It is a set of independent posterior distributions for each of the cells in the grid, which can only be a solution to the inverse problem if it is assumed that $p(\mathbf{g}) = \prod_{i=1}^{M} p(g_i)$ (this can be seen by combining this with equations 1.2 and 1.17 to obtain $p(\mathbf{g}|\hat{\mathbf{e}}) = k \prod_{i=1}^{M} p(g_i) p(\hat{\mathbf{e}}_i|g_i) = k' \prod_{i=1}^{M} p(g_i|\hat{\mathbf{e}}_i)$, where k and k' are normalising constants). This is incompatible with the general definition of geological prior information (section 1.3) which includes spatial correlation between the geological parameters.

Furthermore, an even more serious restriction exists since only one trained neural network is used to invert $\hat{\mathbf{e}}_i$ at each cell (Shahraeeni and Curtis, 2011), this implies that the same $p(g_i)$ distribution is applied within $p(g_i|\hat{\mathbf{e}}_i) \; \forall \; i$. Of course, the neural network could be re-trained for each i with a different $p(g_i)$ but this would obviate the efficiency gains made by using neural network inversion since training is an inherently costly procedure.

Thus, as it is described here, this neural network inversion method is of limited use generally in Bayesian seismic inversion. However, in this thesis we will show that, using Bayes' rule, the cell-wise prior $p(g_i)$ may be efficiently varied within the results of neural network inversion, thus $p(g_i|\hat{\mathbf{e}}_i)$ may be determined with $p(g_i)$ varying with respect to i, without having to retrain the neural network. Furthermore, it will be shown that this so-called prior replacement operation can be used to integrate the neural network-derived estimates of $p(g_i|\hat{\mathbf{e}}_i)$ within stochastic geological inversion, which can incorporate spatial correlation between the parameters g_i (i.e., using a $p(\mathbf{g})$ distribution defined using multi-point geostatistics).

1.8 Outline of the thesis

We now summarise the research topics identified in this introductory chapter and describe how we address them in this thesis. All developments which we make here can be applied to the two-stage workflow as described in section 1.6. Figure 1.1 summarises this workflow, and the modifications which we make to it.

In section 1.6, it was noted, for reasons of computational efficiency, that the elastic inversion part of the two-stage inversion workflow often employs a simple prior distribution $p(\mathbf{e})$, which may contain only a small amount of the available prior

information about $\mathbf{e}$. Thus in Chapter 2 we develop a method which transforms the results of deterministic elastic inversion $\hat{\mathbf{e}}$ (performed using $p(\mathbf{e})$ defined as a Gaussian distribution), using a so-called deep neural network function, such that the new estimates include sophisticated, multi-point geostatistical prior information.

In section 1.7 we described how neural network methods can be used to solve the geological inversion problem. However, these methods are currently only applicable under very restrictive assumptions about the geological prior. In Chapter 3 we develop a so-called *prior replacement* operation using Bayes' rule which relaxes the requirement that $p(g_i)$ be constant with respect to i.

In section 1.6 it was stated that Monte-Carlo techniques for stochastic geological inversion usually generate a correlated set of samples from $p(\mathbf{g}|\hat{\mathbf{e}})$. Because local convergence of the sampler is possible, any estimate made of the geological posterior made using this set of samples is at risk of bias. In Chapter 4 we discuss this problem further, and develop a so-called recursive algorithm which permits exact sampling from the geological posterior distribution, and hence avoids such bias problems. We will also show in that chapter that the prior replacement operation (Chapter 3) permits the use of neural network inversion within this, and other, stochastic geological inversion methods (thus relaxing the requirement that $p(\mathbf{g}) = \prod_{i=1}^{M} p(g_i)$ for the application of neural network inversion to geological inversion).

In section 1.5.2 it was described how training images are used to extract statistics with which the geological prior $p(\mathbf{g})$ can be specified and hence used in inversion. However, appropriate training images often do not exist for a given inversion problem. In this case appropriate training images may be generated by geological experts. However, this process can be costly and difficult. Thus in Chapter 5 we develop a new elicitation method for obtaining the statistics reliably and directly from a geological expert, without the need for training images.

Each of Chapters 2-5 contains a discussion of the method(s) developed therein. Additionally, the implications of these results for Bayesian seismic inversion in general are discussed in Chapter 6. Chapter 7 lists the conclusions that can be made based on the content of this thesis. Appendices A-D contain additional content in support of the main body of work in the thesis.

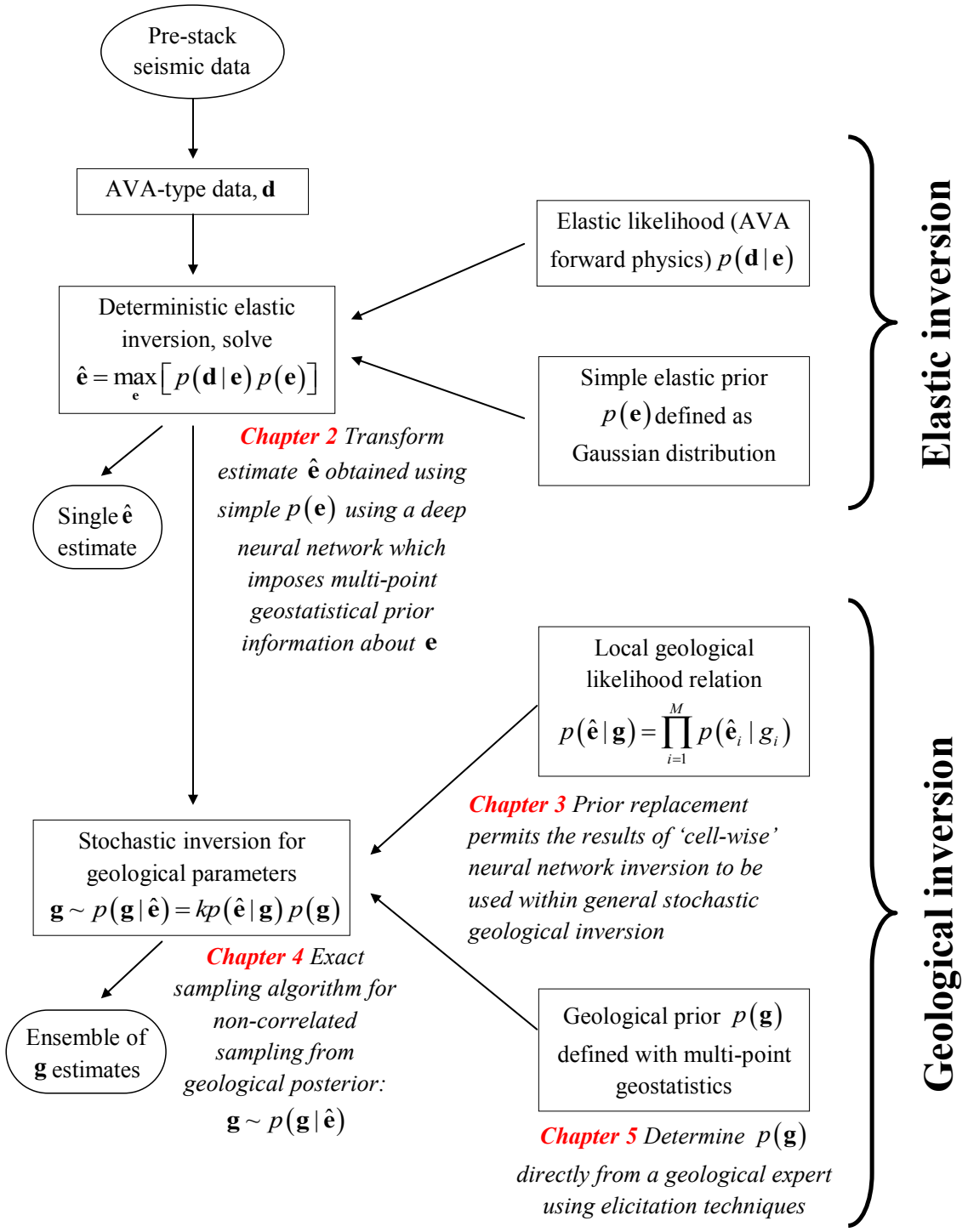

Figure 1.1: The so-called 'two-stage' workflow for Bayesian seismic inversion, which is assumed throughout this thesis and to which we develop improvements. The elements of the workflow which we investigate are annotated in red. Note that k represents the normalising constant in the geological posterior here.

References

Aki, K., and P. G. Richards (2002), *Quantitative seismology*, University Science Books.

Avseth, P., T. Mukerji, and G. Mavko (2005), *Quantitative seismic interpretation*, Cambridge University Press.

Barzilai, A., T. Vanzandt, T. Pike, S. Manionand, and T. Kenny (1998), Improving the performance of a geophone through capacitive position sensing and feedback, in *American Society of Mechanical Engineers International Congress*.

Belisle, C. (1998), Slow convergence of the Gibbs sampler, *Canadian Journal of Statistics*, *26*(4), 629–641.

Besag, J. (1974), Spatial interaction and the statistical analysis of lattice systems, *Journal of the Royal Statistical Society. Series B (Methodological)*, *36*(2), 192–236.

Bo, Y. Y., G. H. Lee, H.-J. Kim, H.-T. Jou, D. G. Yoo, B. J. Ryu, and K. Lee (2013), Comparison of wavelet estimation methods, *Geosciences Journal*, *17*(1), 55–63.

Bosch, M., T. Mukerji, and E. F. Gonzalez (2010), Seismic inversion for reservoir properties combining statistical rock physics and geostatistics: A review, *Geophysics*, *75*(5), 75A165–75A176.

Bosch, M., G. Bertorelli, G. Alvarez, A. Moreno, R. Colmenares, and E. Garcia (2012), Deterministic and Stochastic Seismic Inversion Methods for Gas Discrimination at La Creciente Field, Colombia, in *First EAGE/ACGGP Latin American Geophysics Workshop*.

Buland, A., and H. Omre (2003a), Joint AVO inversion, wavelet estimation and noise-level estimation using a spatially coupled hierarchical Bayesian model, *Geophysical Prospecting*, *51*(6), 531–550.

Buland, A., and H. Omre (2003b), Bayesian linearized AVO inversion, *Geophysics*, *68*(1), 185–198.

Buland, A., and H. Omre (2003c), Bayesian wavelet estimation from seismic and well data, *Geophysics*, *68*(6), 2000–2009.

Caers, J. (2005), *Petroleum geostatistics*, Richardson, TX: Society of Petroleum Engineers.

Caers, J., and T. Hoffman (2006), The probability perturbation method: A new look at Bayesian inverse modeling, *Mathematical Geology*, *38*(1), 81–100.

Caers, J., S. Srinivasan, and A. Journel (1999), Geostatistical quantification of geological information for a fluvial-type North Sea reservoir, in *SPE Annual Technical Conference and Exhibition*.

Castagna, J. P. (1993), AVO analysis - tutorial and review, in *Offset-dependent reflectivity: theory and practice of AVO analysis,* pp.3–36, Society of Exploration Geophysicists.

Chang, C., M. D. Zoback, and A. Khaksar (2006), Empirical relations between rock strength and physical properties in sedimentary rocks, *Journal of Petroleum Science and Engineering*, *51*(3), 223–237.

Cui, H., A. Stein, and D. E. Myers (1995), Extension of spatial information, Bayesian kriging and updating of prior variogram parameters, *Environmetrics*, *6*(4), 373–384.

Curtis, A., and R. Wood (2004), Geological Society of London.

Dubrule, O., M. Thibaut, P. Lamy, and A. Haas (1998), Geostatistical reservoir characterization constrained by 3D seismic data, *Petroleum Geoscience*, *4*(2), 121–128.

Dueholm, K., and T. Olsen (1993), Reservoir analog studies using multimodel photogrammetry: a new tool for the petroleum industry, *AAPG Bulletin*, *77*(12), 2023–2031.

Filippova, K., A. Kozhenkov, and A. Alabushin (2011), Seismic inversion techniques: choice and benefits, *First Break*, *29*(5), 103–114.

Francis, A. (2006), Understanding stochastic inversion: Part 1, *First Break*, *24*(11).

George, E. I., U. Makov, and A. Smith (1993), Conjugate likelihood distributions, *Scandinavian Journal of Statistics,* pp.147–156.

González, E. F., T. Mukerji, and G. Mavko (2007), Seismic inversion combining rock physics and multiple-point geostatistics, *Geophysics*, *73*(1), R11–R21.

Grana, D., T. Mukerji, and J. Dvorkin (2011), Single loop inversion of facies from seismic data using sequential simulations and probability perturbation method, in *2011 SEG Annual Meeting*, Society of Exploration Geophysicists.

Gubbins, D. (2004), *Time series analysis and inverse theory for geophysicists*, Cambridge University Press.

Haas, A., and O. Dubrule (1994), Geostatical inversion - a sequential method of stochastic reservoir modelling constrained by seismic data, *First break*, *12*(11).

Hampson, D. (1991), AVO inversion, theory and practice, *The Leading Edge*, *10*(6), 39–42.

Hampson, D. P., B. H. Russell, and B. P. Bankhead (2005), Simultaneous inversion of pre-stack seismic data, in *2005 SEG Annual Meeting*.

Hill, J., D. Tetzlaff, A. Curtis, and R. Wood (2009), Modeling shallow marine carbonate depositional systems, *Computers & Geosciences*, *35*(9), 1862–1874.

Hubral, P. (1983), Computing true amplitude reflections in a laterally inhomogeneous earth, *Geophysics*, *48*(8), 1051–1062.

Johansson, E. M., F. U. Dowla, and D. M. Goodman (1991), Backpropagation learning for multilayer feed-forward neural networks using the conjugate gradient method, *International Journal of Neural Systems*, *2*(04), 291–301.

Kerry, R., and M. Oliver (2007), Comparing sampling needs for variograms of soil properties computed by the method of moments and residual maximum likelihood, *Geoderma*, *140*(4), 383–396.

Kolbjørnsen, O., M. Stien, H. Kjønsberg, B. Fjellvoll, and P. Abrahamsen (2013), Using multiple grids in Markov Mesh Facies modeling, *Mathematical Geosciences,* pp.1–21.

Lamy, P., P. Swaby, P. Rowbotham, O. Dubrule, and A. Haas (1999), From seismic to reservoir properties with geostatistical inversion, *SPE Reservoir Evaluation & Engineering*, *2*(04), 334–340.

Mavko, G., T. Mukerji, and J. Dvorkin (2009), *The rock physics handbook: Tools for seismic analysis of porous media*, Cambridge University Press.

Mosegaard, K., and M. Sambridge (2002), Monte Carlo analysis of inverse problems, *Inverse Problems*, *18*(3), R29.

Mukerji, T., P. Avseth, G. Mavko, I. Takahashi, and E. F. González (2001), Statistical rock physics: Combining rock physics, information theory, and geostatistics to reduce uncertainty in seismic reservoir characterization, *The Leading Edge*, *20*(3), 313–319.

Olea, R. (1999), *Geostatistics for engineers and earth scientists*, Kluwer Academic Boston.

Pendrel, J. (2001), Seismic inversion - the best tool for reservoir characterization, *CSEG Recorder*, *26*(1), 18–24.

Pham, T. D. (2010), GeoEntropy: A measure of complexity and similarity, *Pattern Recognition*, *43*(3), 887–896.

Pringle, J., A. Westerman, J. Clark, N. Drinkwater, and A. Gardiner (2004), 3D high-resolution digital models of outcrop analogue study sites to constrain reservoir model uncertainty: an example from Alport Castles, Derbyshire, UK, *Petroleum Geoscience*, *10*(4), 343–352.

Remy, N., A. Boucher, and W. Jianbing (2009), Applied geostatistics with SGeMS: a user's guide, *The Leading edge*, *28*(12).

Rimstad, K., P. Avseth, and H. Omre (2012), Hierarchical Bayesian lithology/fluid prediction: A North Sea case study, *Geophysics*, *77*(2), B69–B85.

Ringrose, P., G. Pickup, J. Jensen, and M. Forrester (1999), The Ardross reservoir gridblock analog: sedimentology, statistical representivity, and flow upscaling, in *AAPG Memoir 71: Reservoir Characterization-Recent Advances*, edited by R. A. Schatzinger and J. F. Jordan, pp.265–276, AAPG.

Roth, G., and A. Tarantola (1994), Neural networks and inversion of seismic data, *Journal of Geophysical Research: Solid Earth*, *99*(B4), 6753–6768.

Sambridge, M., K. Gallagher, A. Jackson, and P. Rickwood (2006), Trans-dimensional inverse problems, model comparison and the evidence, *Geophysical Journal International*, *167*(2), 528–542.

Sava, P. C., and S. Fomel (2003), Angle-domain common-image gathers by wavefield continuation methods, *Geophysics*, *68*(3), 1065–1074.

Shahraeeni, M. S., and A. Curtis (2011), Fast probabilistic nonlinear petrophysical inversion, *Geophysics*, *76*(2), E45–E58.

Shahraeeni, M. S., A. Curtis, and G. Chao (2012), Fast probabilistic petrophysical mapping of reservoirs from 3D seismic data, *Geophysics*, *77*(3), O1–O19.

Shuey, R. (1985), A simplification of the Zoeppritz equations, *Geophysics*, *50*(4), 609–614.

Spikes, K., T. Mukerji, J. Dvorkin, and G. Mavko (2007), Probabilistic seismic inversion based on rock-physics models, *Geophysics*, *72*(5), R87–R97.

Srivastava, R., and M. Sen (2010), Stochastic inversion of prestack seismic data using fractal-based initial models, *Geophysics*, *75*(3), R47–R59.

Thore, P. (2013), Data driven versus model based inversion - When and why?, in *75th EAGE Conference Exhibition extended abstracts*, EAGE.

Torres-Verdin, C., M. Victoria, G. Merletti, and J. Pendrel (1999), Trace-based and geostatistical inversion of 3-D seismic data for thin-sand delineation: An application in San Jorge Basin, Argentina, *The Leading Edge*, *18*(9), 1070–1077.

Truong, P. N., G. Heuvelink, and J. P. Gosling (2013), Web-based tool for expert elicitation of the variogram, *Computers & Geosciences*, *51*, 390–399.

Tsvankin, I., J. Gaiser, V. Grechka, M. van der Baan, and L. Thomsen (2010), Seismic anisotropy in exploration and reservoir characterization: An overview, *Geophysics*, *75*(5), 75A15–75A29.

Ulrych, T. J., M. D. Sacchi, and A. Woodbury (2001), A Bayes tour of inversion: A tutorial, *Geophysics*, *66*(1), 55–69.

Virieux, J., and S. Operto (2009), An overview of full-waveform inversion in exploration geophysics, *Geophysics*, *74*(6), 1–26.

Zhang, R., M. K. Sen, S. Phan, and S. Srinivasan (2012), Stochastic and deterministic seismic inversion methods for thin-bed resolution, *Journal of Geophysics and Engineering*, *9*(5), 611.

Chapter 2

Improving elastic inversion results using deep neural networks

2.1 Overview

In section 1.6 we described how, for reasons of computational efficiency, most elastic inversion methods do not implement all available prior information about $\mathbf{e}$. Thus in this chapter we develop a methodology which aims to transform the estimates of $\mathbf{e}$ made using deterministic elastic inversion $\hat{\mathbf{e}}$, which are constrained only by a simple two-point geostatistical prior model (a Gaussian), to higher resolution estimates containing sophisticated multi-point statistical prior information.

2.2 Introduction

Roughly speaking, our method seeks to learn the mapping, using a neural network function, between the results of deterministic elastic inversion, denoted $\hat{\mathbf{e}}$, and the true earth elastic parameters, denoted $\mathbf{e}$. Such a mapping can then be applied to $\hat{\mathbf{e}}$, obtained by inverting real AVA-type data $\mathbf{d}$ collected over a region of interest, to yield an estimate of $\mathbf{e}$ for that region. To learn such a mapping we first specify a prior probability density function (PDF) $p(\mathbf{e})$, which accurately represents our prior knowledge of $\mathbf{e}$. This PDF is used to generate a large number of possible realisations of $\mathbf{e}$. Corresponding AVA-type data $\mathbf{d}$ is then generated using the AVA forward physics (section 1.4), and this is then inverted using deterministic inversion to obtain estimates $\hat{\mathbf{e}}$. Thus a set of synthetic 'example' pairs of $\mathbf{e}$ and $\hat{\mathbf{e}}$ is obtained,

and this constitutes a so-called training dataset. This dataset is used to estimate the parameters of a neural network function which cause that function to emulate the desired mapping $\hat{\mathbf{e}} \rightarrow \mathbf{e}$.

Neural networks were introduced in section 1.7 for solving the geological inverse problem repeatedly, in isolation at single cells in the model grid. As described there, a neural network is a function which can be used to emulate any mapping between an input and an output variable (Bishop, 1995). Similar previous applications of neural networks to AVA-type data have applied limited prior geological information to inversion of zero-offset seismic data for facies classification (Caers, 2001). Recent developments in neural network theory may now allow us to improve upon such results; so-called deep neural networks have become feasible to train for regression problems (Hinton and Salakhutdinov, 2006; Parviainen, 2010). These network functions have a more complex topology than networks used previously (e.g., Caers, 2001; Shahraeeni et al., 2012), which permits mappings to be learned more efficiently (Erhan et al., 2010) and with less sensitivity to noise in the input data (Vincent et al., 2010).

Importantly, the deterministic elastic inversions required in our method need not contain all of the prior information which is available since accurate prior information is applied by the neural network, which is trained to apply the prior information in the training dataset. Thus we are free to use a simple (henceforth 'low-fidelity') prior distribution (such as Gaussian) for deterministic elastic inversion, which need only permit efficient and stable elastic inversion. Furthermore, to construct a training dataset for the neural network we need only be able to sample from a (henceforth 'high-fidelity') prior distribution which accurately represents our prior knowledge about $\mathbf{e}$: the PDF does not need to be constructed explicitly/parametrically, but may nevertheless contain sophisticated (multi-point geostatistical) prior information. For clarity, we discuss the notation used in the rest of this chapter below, before an outline of the practical implementation of the new method, and the rest of this chapter, is given in section 2.4.

2.3 Notation

It should be noted that the definition of $\mathbf{e}$ here as the *true* earth elastic parameters is consistent with the Bayesian definition of this vector in Chapter 1 (section 1.2) as a random vector. Strictly-speaking, however, it does imply that the mapping we

obtain should be probabilistic, which is to say we should obtain the mapping between the deterministic elastic inversion results and the elastic posterior *distribution* (i.e., $\hat{\mathbf{e}} \rightarrow p(\mathbf{e}|\mathbf{d})$). This is not a trivial task, and in the following we will actually derive a mapping which is approximate in a number of respects.

One of these approximations is that the neural network, written $\mathbf{q}$, is trained to emulate a certain mapping which is applied recursively down a single trace of the deterministic elastic parameter estimates located at a given lateral position $\mathbf{x} = [x, y]$ in a subsurface model grid. Thus we consider a strictly one-dimensional *recursive operation* which is unable to enforce information about the lateral variation of the elastic parameters (e.g., about lateral correlation). Ideally, we would define $\mathbf{q}$ to act as a three-dimensional (3-D) recursive operator which can apply such information to a 3-D grid, however we have chosen the 1-D limitation to reduce the computational costs and practical difficulties associated with training deep neural networks (Bengio, 2012).

Since the the operation which we apply is 1-D, we will only consider a 1-D grid in the derivation of $\mathbf{q}$. Thus in this chapter, the vectors $\mathbf{e}$, $\mathbf{d}$ and $\hat{\mathbf{e}}$ now represent quantities down a single trace (i.e., down the z direction) only, or equivalently $\mathbf{e} = \mathbf{e}_{\mathbf{x}}$, $\mathbf{d} = \mathbf{d}_{\mathbf{x}}$ and $\hat{\mathbf{e}} = \hat{\mathbf{e}}_{\mathbf{x}}$ where $\mathbf{x} = [x = 1, y = 1]$. Thus the notation $\mathbf{e}_z$ is used to refer to an elastic parameter vector at the cell with vertical coordinate (or equivalently index) equal to z, and $\mathbf{e}_i^j = [\mathbf{e}_i, \mathbf{e}_{i+1}, ..., \mathbf{e}_j]$ is used to represent all elastic parameter vectors (in a single trace) in cells with z coordinates (or indices) between i and j (inclusive). This notation also applies to $\hat{\mathbf{e}}$ and $\mathbf{d}$. However, we will demonstrate the method by repeatedly applying the 1-D operation to all traces at different lateral positions within a real 3-D grid of data, at which point we shall use the $\mathbf{x}$ coordinate to differentiate between lateral positions (i.e., we will apply the operation to $\hat{\mathbf{e}}_{\mathbf{x}}$ at all $\mathbf{x}$ positions in the grid).

Otherwise, in the most part, the notation used in this chapter is in agreement with that used in Chapter 1, and where deviations exist they are noted in the text. However, a summary of the notation used in this chapter is provided in Appendix H.1 for reference.

2.4 Outline of the method

To outline the method we begin by supposing that we have obtained some AVA-type data $\mathbf{d}^r$ corresponding to the true elastic parameters down a trace at some lateral position, which we refer to as $\mathbf{e}^r$, where the superscript r is henceforth used to denote real (as opposed to synthetic) quantities. Our method of estimation for $\mathbf{e}^r$ (i.e., including the multi-point geostatistical information) then comprises the following 10 steps (see Figure 2.1):

(1) Define a 'low-fidelity' prior: Define a prior PDF $p_L(\mathbf{e})$ which is mono-modal and cheap to evaluate and thus promotes computationally efficient elastic inversion (i.e., a Gaussian).

(2) Deterministic inversion of real AVA-type data: Perform deterministic inversion of $\mathbf{d}^r$ using the low-fidelity prior $p_L(\mathbf{e})$. The results are written $\hat{\mathbf{e}}^r$, where $\hat{}$ denotes an (MAP) estimate and r implies that it is made using the real data.

(3) Define a 'high-fidelity' prior: Define a prior PDF $p_H(\mathbf{e})$ which accurately represents our prior knowledge of $\mathbf{e}$, where ideally $\mathbf{e}^r \sim p_H(\mathbf{e})$. This PDF need not be constructed parametrically, since it need only be sampled from in step 4.

(4) Generate synthetic elastic realisations: Sample from $p_H(\mathbf{e})$ to generate B trace realisations of the elastic parameters, $\mathbf{e}^s \sim p_H(\mathbf{e})$, where the superscript s implies that these realisations are synthetic.

(5) Generate synthetic AVA-type data: Using the forward physics and the B $\mathbf{e}^s$ trace realisations from step 4 generate the corresponding B synthetic AVA-type data $\mathbf{d}^s = \mathbf{f}(\mathbf{e}^s)$ traces.

(6) Deterministic inversion of synthetic AVA-type data: Invert all B synthetic data $\mathbf{d}^s$ traces from step 5 deterministically to obtain B traces of elastic parameter estimates $\hat{\mathbf{e}}^s$. For consistency, the deterministic inversions here use the same $p_L(\mathbf{e})$ from steps 1 and 2.

(7) Form the training dataset: From the B pairs of $\mathbf{e}^s$ (step 4) and $\hat{\mathbf{e}}^s$ (step 6) traces extract the training dataset (pairs of input and output) which will be used to train $\mathbf{q}$ to emulate the desired mapping (to be used in the 1-D recursive operation).

(8) Define the neural network q: Choose a suitable topology (i.e., parametrisation) for $\mathbf{q}$ which will allow it emulate the desired mapping (to be used in the 1-D recursive operation).

(9) Train the neural network q: Using the training dataset formed in step 7, train **q** (step 8) to emulate the mapping which upon its recursive application to $\hat{\mathbf{e}}$ will approximately do the transformation $\hat{\mathbf{e}} \rightarrow \mathbf{e}$.

(10) Apply the recursive operation using q: Apply **q** (trained in step 9) within the 1-D recursive operation to do the approximate transformation $\hat{\mathbf{e}} \rightarrow \mathbf{e}$.

In the following sections we describe each step of the methodology in greater detail. We have already described the forward physics relating the AVA-type data to the subsurface elastic parameters (steps 2, 5 and 6) in section 1.4. However, in section 2.5 we will describe the specific deterministic inversion procedure using the low-fidelity prior (steps 1, 2 and 6) in detail. We then precisely define the 1-D recursive operation, and the particular mapping within this which is emulated by the neural network function **q** in section 2.6. We then discuss neural networks in general and the training and topology of **q** specifically (steps 7, 8, 9 and 10) in section 2.7.

It must be noted that although the method developed here is strictly 1-D, this does not mean that the recursive operator is not useful for 3-D (or 2-D) grids; the same recursive operation may be applied repeatedly at different lateral positions $\mathbf{x}$ within a (2-D or 3-D) grid, so long as the (1-D) prior information applied by the operator is valid $\forall$ $\mathbf{x}$ in the grid. Thus, in section 2.8 we apply the methodology (steps 1-10) to real data where we provide an example of how a high-fidelity prior PDF (steps 3 and 4) can be constructed, and sampled-from, for a given geological-setting. The real data comprises a large 3-D grid of data, where at each lateral position (i.e., trace) in the grid we apply the same 1-D recursive operation.

2.5 Deterministic seismic inversion

In order to evaluate the elastic posterior in equation 1.15, and hence perform deterministic inversion, we must define the elastic prior PDF. A simple low-fidelity prior PDF $p_L(\mathbf{e})$ is used for deterministic inversion here in order to facilitate efficient deterministic elastic inversion. Thus we choose a Gaussian PDF defined as

$$p_L(\mathbf{e}) = (2\pi)^{-\frac{k}{2}} |\mathbf{\Sigma_e}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(-(\mathbf{e}-\mathbf{e_0})^T \mathbf{\Sigma_e}^{-1} (\mathbf{e}-\mathbf{e_0})\right)\right) \tag{2.1}$$

where $\mathbf{e_0}$ is the initial (or mean) model, $\mathbf{\Sigma_e}$ is the prior covariance matrix and k is the dimensionality of the $\mathbf{e}$ vector. $\mathbf{\Sigma_e}$ describes the prior spatial correlation of the

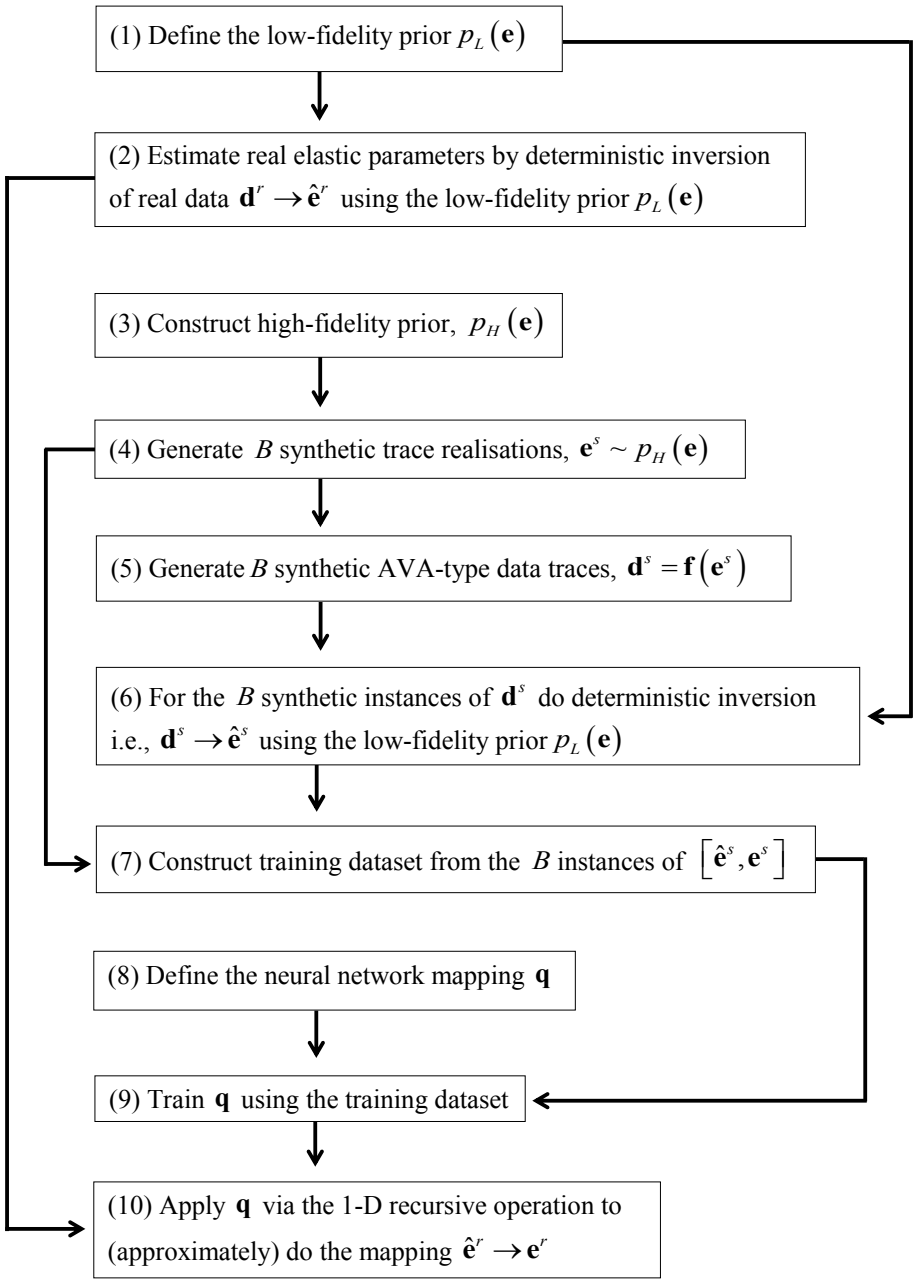

Figure 2.1: Outline of the methodology for estimating $\mathbf{e}^r$ from $\mathbf{d}^r$, where numbers refer to the chronological order and arrows imply dependence between steps.

$\mathbf{e}_z$ variables vertically down a trace at a single lateral location. It can be constructed from a variogram model. A suitable low-fidelity choice for $\mathbf{e}_0$ is a low frequency model (i.e., a model which expresses only large-scale, general depth-trends in the subsurface). This can be obtained by applying a low-pass filter to local well-log data of $\mathbf{e}$ if available, or by using known regional depth trends. Similarly $\mathbf{\Sigma_e}$ can be constructed using an empirical variogram calculated from well-log measurements of $\mathbf{e}$.

In deterministic inversion we aim to find the MAP value by maximising equation 1.15. Substituting equations 1.6 and 2.1 into equation 1.15, it can be shown that this is equivalent to the minimisation

$$\hat{\mathbf{e}} = \arg\min_{\hat{\mathbf{e}}} \left(\|\mathbf{f}(\hat{\mathbf{e}}) - \mathbf{d}(\mathbf{e})\|_{\mathbf{\Sigma_d}} + \|\hat{\mathbf{e}} - \mathbf{e}_0\|_{\mathbf{\Sigma_e}} \right), \tag{2.2}$$

where $\hat{\mathbf{e}}$ and $\mathbf{e}$ are the estimated and true elastic parameters, respectively, and the notation $\|\mathbf{v}\|_{\mathbf{C}} = \mathbf{v}^T\mathbf{C}^{-1}\mathbf{v}^T$. Note that in the above equation $\mathbf{d}(\mathbf{e})$ should be interpreted as the observed AVA-type data. It can be shown (see e.g., Gubbins, 2004) that the estimate may be written as the matrix multiplication of the true elastic parameters and the so-called resolution matrix as

$$\hat{\mathbf{e}} = (\mathbf{A}^T\mathbf{\Sigma_d^{-1}}\mathbf{A} + \mathbf{\Sigma_e^{-1}}(\mathbf{A}^T\mathbf{\Sigma_d^{-1}}\mathbf{A}))\mathbf{e}. \tag{2.3}$$

where $\mathbf{A} = \mathbf{S}\dot{\mathbf{R}}(\hat{\mathbf{e}})$, in which $\mathbf{S}$ is the wavelet block matrix (see section 1.4) and $\dot{\mathbf{R}}(\hat{\mathbf{e}})$ is a matrix containing the derivatives of the reflectivity vector $\mathbf{R}$ (see section 1.4) with respect to $\mathbf{e}$, evaluated at $\hat{\mathbf{e}}$. From the resolution matrix we see that each element of the estimate vector $\hat{\mathbf{e}}$ is a linear combination of a number of elements of the true $\mathbf{e}$ vector. The presence of $\mathbf{S}$ in equation 2.3 implies that the wavelet vectors have a strong influence on the vertical range of this linear combination. Generally speaking, this means that elastic parameters which are close together down a trace are harder to resolve than those further apart.

2.6 The recursive operation

We now define the recursive 1-D operation applied, and the particular mapping within this which is approximated by the neural network function $\mathbf{q}$. It is well-known that the resolution matrix in equation 2.3 cannot be inverted since there

is not a unique mapping between the estimates $\hat{\mathbf{e}}$ and the true elastic parameters $\mathbf{e}$ (which in the Bayesian interpretation is a random vector). We can nevertheless write the relationship as a probability distribution, $p(\mathbf{e}|\hat{\mathbf{e}})$, which can be rewritten using conditional probability distributions as

$$p(\mathbf{e}|\hat{\mathbf{e}}) = \prod_{z=1}^{T} p(\mathbf{e}_z|\mathbf{e}_1^{z-1}, \hat{\mathbf{e}}). \tag{2.4}$$

where the notation $\mathbf{e}_i^j = [\mathbf{e}_i, \mathbf{e}_{i+1}, ..., \mathbf{e}_j]$ is used. We can approximate equation 2.4 by limiting the dependency within the conditional probability distributions on the right hand side to some characteristic depth-lag, denoted λ, giving

$$p(\mathbf{e}|\hat{\mathbf{e}}) \approx \prod_{z=1}^{T} p(\mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}). \tag{2.5}$$

Dependency in the resolution matrix (equation 2.3) is controlled by the wavelet vectors specified in $\mathbf{S}$. Furthermore, we assume henceforth that the effective range of geological correlation is less than the wavelength of the wavelet. Thus we propose that a reasonable choice for λ is to set it equal to half of the period of the longest of the angle dependent wavelets $[\mathbf{w}_{near}, \mathbf{w}_{mid}, \mathbf{w}_{far}]$, and we use this approximation henceforth.

Equation 2.5 can be sampled from using sequential sampling for $z = 1, 2, ..., Z$; such sampling can therefore be thought of as a recursive operator. Much work has been done on using neural networks to determine probabilistic mappings like $[\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}] \rightarrow p(\mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda})$ in equation 2.5 (e.g., Bishop, 1994; Barber and Bishop, 1998). However, such methods typically require networks with very large numbers of free parameters (in order to characterise the posterior distribution over the full extent of the parameter space). Therefore in order to ease the computational burden of training we do not consider determining the full probability distribution. Instead we use the neural network function $\mathbf{q}$ to emulate a related *injective* mapping, defined as

$$[\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}] \rightarrow \mathrm{E}[\mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}]. \tag{2.6}$$

where E[] is the expectation operator. Then the 1-D recursive operation is defined by calculating this mapping (equation 2.6) for $z = 1, 2, ..., Z$, where after calculation at

z we set $\mathbf{e}_z = \mathrm{E}[\mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}]$, as demonstrated in Figure 2.2. Not all of the required $\hat{\mathbf{e}}_z$ or $\mathbf{e}_z$ input values may exist at the beginning or end of a trace, where $z-\lambda < 0$ or $z+\lambda > Z$ respectively (see Figure 2.2). For such positions we train neural networks to emulate mappings with modified topology to accommodate the 'missing' inputs (we do not explain these networks further since they are obtained and applied in the same way as $\mathbf{q}$).

Note that each dependency (i.e., arrow) in Figure 2.2 describes the dependency of all three elastic parameters at depth z on all three of the elastic parameters at the conditioning depths. Thus the values of $[I_P, I_S, \rho]$ are predicted simultaneously and should be consistent with one another at depth z. Additionally, the recursive application of $\mathrm{E}[\mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}]$ down a trace is not equivalent to calculating $\mathrm{E}[\mathbf{e}|\hat{\mathbf{e}}]$, because the directional nature of its application will mean that each estimate of $\mathbf{e}_z$ is inherently biased. Nevertheless, it can be used to ensure that a geologically reasonable sample is obtained because the conditional expectation ensures vertical spatial dependency between the elastic parameter estimates. This is in contrast to what would be obtained if we instead estimated $\mathrm{E}[\mathbf{e}|\hat{\mathbf{e}}]$; in this case each elastic parameter estimate at each z could be determined independently of all others following the laws of expectations, and thus geological continuity would not be ensured.

For later convenience we define two vectors $\mathbf{u} = [\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}]$ and $\mathbf{v} = \mathbf{e}_z|\mathbf{e}_{z-\lambda}^{z-1}, \hat{\mathbf{e}}_{z-\lambda}^{z+\lambda}$, such that $\mathbf{u} \rightarrow \mathrm{E}[\mathbf{v}]$. Since $\mathbf{e}_z$ and $\hat{\mathbf{e}}_z$ each have three elements, the number of elements in $\mathbf{u}$ and $\mathbf{v}$ are

$$3 \times (\lambda + (1 + (2 \times \lambda)) \text{ and } 3, \tag{2.7}$$

respectively. Given the definition of $\mathbf{u}$ and $\mathbf{v}$, the pair of vectors $[\hat{\mathbf{e}}^s, \mathbf{e}^s]$ for a single trace, can yield numerous realisations of pairs of these vectors, which we write $[\mathbf{u}^s, \mathbf{v}^s]$ where the s superscript indicates that this is synthetic data. Thus for convenience we define the operation $\mathcal{Q}$, which extracts

$$Z - 2\lambda \tag{2.8}$$

instances of the $[\mathbf{u}^s, \mathbf{v}^s]$ pair from a single $[\hat{\mathbf{e}}^s, \mathbf{e}^s]$ pair of traces (where the number of instances which may be extracted from a single trace is limited by the size of λ because of the finite length of traces, as discussed above). These vector pairs will be used to form the training dataset for the neural network mapping.

As stated above we use a neural network $\mathbf{q}$ to approximate equation 2.6. Thus $\mathbf{u}^s$ constitutes a realisation of the input of $\mathbf{q}$ but $\mathbf{v}^s$ is not a realisation of the desired output variable, which is the expectation $\mathrm{E}[\mathbf{v}^s]$. However, we will show later that a training dataset comprising pairs of $[\mathbf{u}^s, \mathbf{v}^s]$ is sufficient to induce the neural network to emulate the desired mapping $\mathbf{q} : \mathbf{u} \rightarrow \mathrm{E}[\mathbf{v}]$.

2.7 Neural networks

2.7.1 Topology of neural networks

A neural network is a function whose structure, when expressed graphically, is similar to the physical arrangement of biological neurons. They comprise $L + 1$ layers of nodes within which each node is connected by edges (connecting lines) to all of the nodes in directly adjacent layers, but there are no edges between nodes within a layer. Each layer has K^l variable nodes where $l \in [0, ..., L]$ refers to the layer number (it is not an exponent). Each node is associated with a variable a_i^l, which is the variable associated with the i^{th} node of the l^{th} layer. Each edge is associated with a weight w_{ij}^l, which is the weight associated with the edge connecting the i^{th} node of the $l - 1^{th}$ layer to the j^{th} node of the l^{th} layer. Figure 2.3 illustrates such a neural network structure.

Additionally, each layer contains a so-called bias node, which is the zeroth node in a layer and is associated with variables a_0^l and weights w_{0j}^l (thus the total number of nodes in a layer is $K^L + 1$). Bias nodes serve only to supply a weighted constant to nodes in the layer above, thus they do not have any connections to the nodes in the layer below, and their associated variables are constant, i.e., $a_0^l = 1 \; \forall \; l \; \in \; [0, ..., L]$. Thus note that, by definition, the bias node in $l = L$ is redundant and is henceforth ignored. The vector notation $\mathbf{a}^l = [a_0^l, ..., a_{K^l}^l]$ is used to denote the set of all variables (nodes, including the bias) in layer l. A shorthand can be used to describe the number of layers and nodes (including biases) in a network, e.g., $3 - 3 - 4 - 1$ describes the network in Figure 2.3.

By definition, information in a neural network passes 'upward' only from the so-called input layer $l = 0$ to the output layer $l = L$. All intermediate layers are referred to as hidden layers. The variables (except the constant biases) in each layer (except the input) can be defined as a function of the weighted variables (including

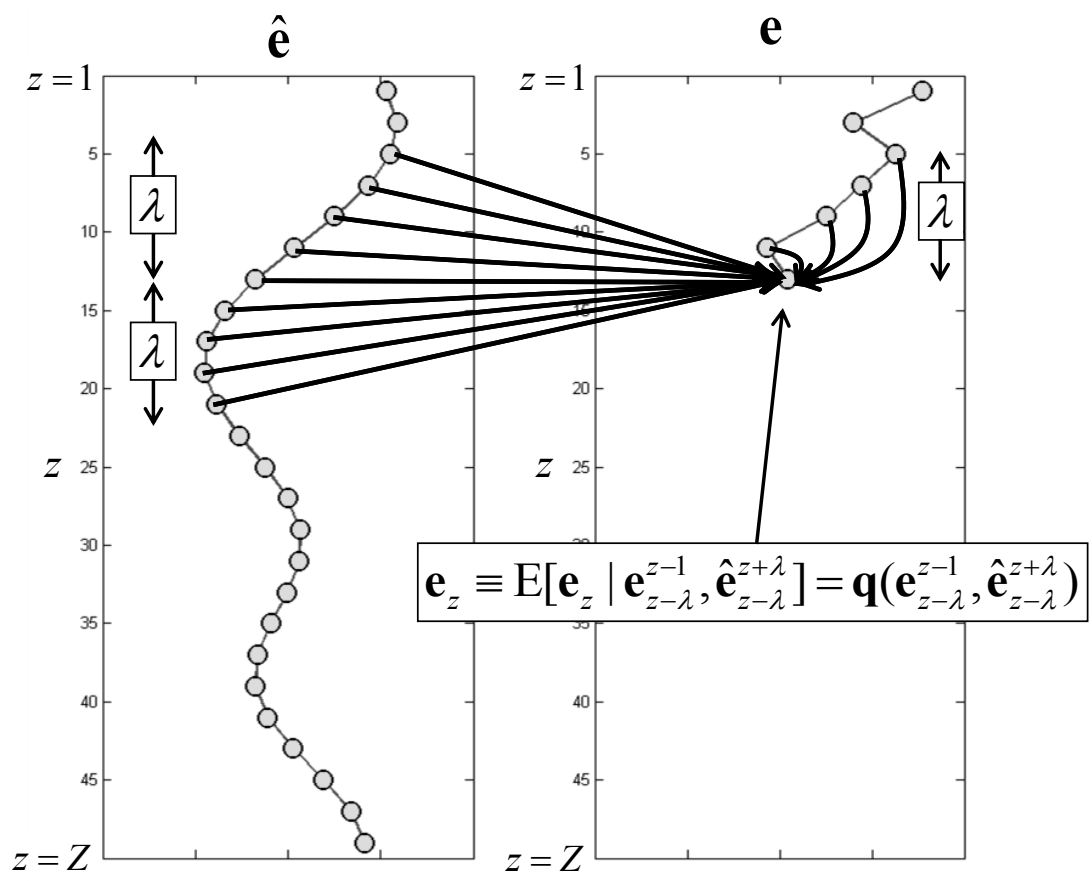

Figure 2.2: The recursive operation applies a mapping, approximated by the neural network function $\mathbf{q}$, recursively for $z = 1, ..., Z$ down a trace at a given lateral position to predict $\mathbf{e}_z \; \forall \; z$. At each z, $\mathbf{q}$ returns the expected value for $\mathbf{e}_z$ given the results of deterministic seismic inversion $\hat{\mathbf{e}}$ and the previously predicted $\mathbf{e}_z$ values down the trace. $\mathbf{e}_z$ is then set to the expected value, such that it is used as input to $\mathbf{q}$ for predicting $\mathbf{e}_{z+1}$ and so forth. The vertical dependency of $\mathbf{q}$ is limited to λ cells/samples above (for $\mathbf{e}$ and $\hat{\mathbf{e}}$) and below (for $\hat{\mathbf{e}}$) the current cell/sample, z.

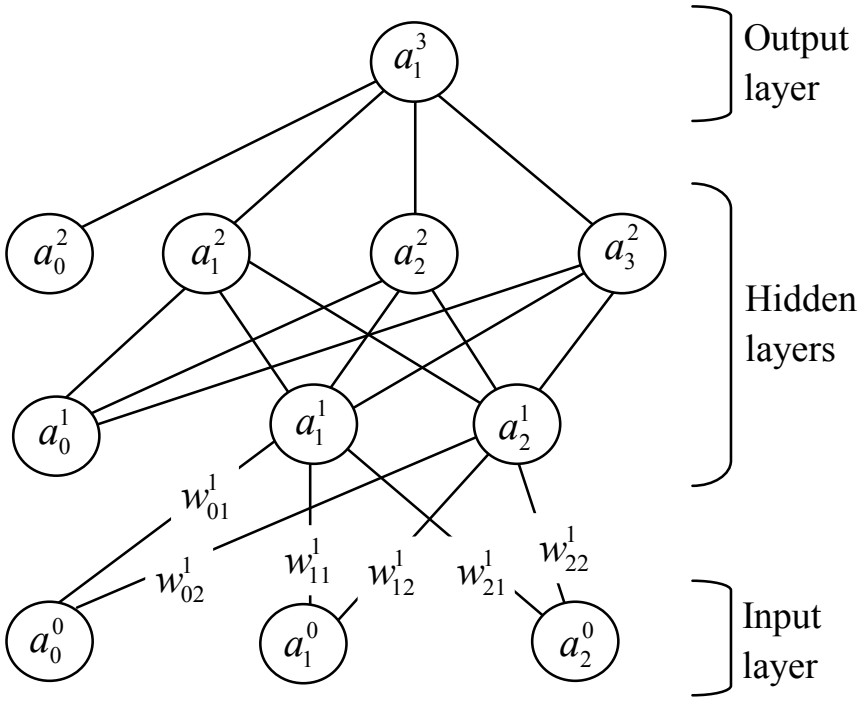

Figure 2.3: A neural network with 2 layers of hidden nodes. In total there are 4 layers of nodes (thus $L = 3$). The i^{th} node of the l^{th} layer is associated with the variable a_i^l, where $l \in [0, ..., L]$ is the layer index. Edges (connecting lines) connect the i^{th} node in the $(l-1)^{th}$ layer with the j^{th} node in the l^{th} and are associated with a weight w_{ij}^l. All layers of nodes have an additional bias node ($i = 0$) whose associated variable is set constant (i.e., $a_0^l = 1 \; \forall \; l \in [0, ..., L]$) and thus has no connecting edge to the nodes in the layer below. Note that in the output layer ($l = L$) the bias node is redundant and is ignored (see equation 2.9).

the biases) in the layer directly 'below', that is

$$a_j^l = g\left(\sum_{i=0}^{K^{l-1}} w_{ij}^l a_i^{l-1}\right) \;\; \forall\; l \in [1, ..., L]\;,\;\; j \in [1, ..., K^l], \tag{2.9}$$

where the summation is over the $K^{l-1}+1$ nodes in the $(l-1)^{th}$ layer, which connect to the j^{th} node in the l^{th} layer. The above applies for all layers except the input layer for which we must supply independent values (i.e., the input vector) for all the variables except the constant bias node. For example, in our case the input vector is $\mathbf{u}$, thus we write $\mathbf{a}^0_{\setminus 0} = \mathbf{u}$, where the $\setminus 0$ subscript implies the set of all variables in the first layer except the bias. The neural network function then predicts the output vector $\mathbf{a}^L$, by calculating equation 2.9 for each layer successively, until the output layer L. Thus ultimately the variables in the output layer are a function of the input variables and the weights, hence we may write our neural network function as $\mathbf{q}(\mathbf{u}; \mathbf{W}) = \mathbf{a}^L(\mathbf{u}; \mathbf{W})$, where $\mathbf{W}$ is a matrix storing the weights' values as $W_{i,j,l} = w_{ij}^l$.

The function g in equation 2.9 is the so-called activation function (Bishop, 1995). This could be different for every node but we assume that all such functions are the sigmoid function, defined as

$$g(x) = \frac{1}{1 + e^{-x}} \tag{2.10}$$

for a scalar input x. It can be shown that any function can be approximated to arbitrary accuracy with a neural network with sigmoidal activation functions with at least one layer of hidden nodes, and a sufficient number of nodes in those hidden layers (see e.g., Bishop, 1995, pp.128-132). In general, the more complex the function the greater the number of hidden nodes required to emulate that mapping.

2.7.2 Training of neural networks

We wish to obtain the neural network such that $\mathbf{q}(\mathbf{u}; \mathbf{W}) : \mathbf{u} \rightarrow \mathrm{E}[\mathbf{v}]$. Suppose that we have a network with certain topology, i.e., number of hidden layers and nodes within those layers (the size of the input and output layer is dictated by the length of the $\mathbf{u}$ and $\mathbf{v}$ vectors, respectively). To induce such a network to emulate the desired mapping we must obtain *appropriate* values for this network's weights $\mathbf{W}$ via training. This uses the training dataset, which in this case is a set

of N 'example' pairs of $\mathbf{u}$ and $\mathbf{v}$ drawn from the joint distribution $p(\mathbf{u}, \mathbf{v})$, written $[\mathbf{u}_i^s, \mathbf{v}_i^s]$, $i = 1, 2, ..., N$. As explained above we can use the operator $\mathcal{Q}$ to obtain such pairs from $[\hat{\mathbf{e}}^s, \mathbf{e}^s]$ pair(s). We can then estimate appropriate values for $\mathbf{W}$ by minimising a sum-of-squares error function, defined as

$$E_N = \sum_{i=1}^{N} (\mathbf{v}_i^s - \mathbf{q}(\mathbf{u}_i^s; \mathbf{W}))^2, \tag{2.11}$$

with respect to $\mathbf{W}$, where it should be understood that $\mathbf{v}^s$ is a sample of $\mathbf{v}$ from the training dataset, whereas $\mathbf{q}(\mathbf{u}_i^s; \mathbf{W})$ is the output of the neural network for a given set of weights $\mathbf{W}$, and input equal to the corresponding (i^{th}) input vector $\mathbf{u}_i^s$ in the training dataset. For a given network topology and training dataset, minimising equation 2.11 yields the maximum likelihood value for $\mathbf{W}$. This minimisation is performed using iterative gradient-descent where the gradients (with respect to $\mathbf{W}$) are calculated with the so-called back-propagation technique (Appendix B).

It can be shown (Bishop, 1995) that if we have $N = \infty$ and a suitably large number of hidden nodes in our network, then minimisation of equation 2.11 yields a set of weights $\mathbf{W}$ which induce the neural network to output the mapping $\mathbf{q}(\mathbf{u}; \mathbf{W}) : \mathbf{u} \rightarrow \mathrm{E}[\mathbf{v}]$ exactly. In practice, the training dataset consists of a finite number of $[\mathbf{u}, \mathbf{v}]$ pairs, limited by the amount of data which can be feasibly created, and incorporated into training. Furthermore, choosing a 'suitably large' number of hidden nodes is not a trivial problem: there must be enough nodes to give the neural network model sufficient flexibility to fit the variation in $\mathrm{E}[\mathbf{v}]$, but not so much as to induce the neural network to fit noise (or stochastic variation) in the training data (i.e., 'over-fit'). In the next section we will consider how these two problems may be mitigated and how $\mathbf{W}$ should be determined in practice.

2.7.3 Generalisation

Since we cannot have $N = \infty$ it is necessary for us to consider the *generalisation* of the neural network, which refers to the ability of $\mathbf{q}(\mathbf{u}; \mathbf{W})$ to correctly predict $\mathrm{E}[\mathbf{v}]$ for a $\mathbf{u}$ vector which was not in the training dataset (Bishop, 1995, p.2). Generally speaking, $\mathbf{q}(\mathbf{u}; \mathbf{W})$ will perform the mapping relatively poorly for $\mathbf{u}$ vectors which are distant from the input vectors in the training dataset used to determine $\mathbf{W}$ (and vice-versa). The error surface in equation 2.11 may contain many local minima to which gradient-descent may converge, and furthermore the value of equation 2.11 for some

$\mathbf{W}$ is not directly proportional to the generalisation performance of $\mathbf{q}(\mathbf{u};\mathbf{W})$, i.e., a lower training data misfit does not necessarily guarantee better generalisation. Thus in practice we must be able to quantify the generalisation performance of $\mathbf{q}(\mathbf{u};\mathbf{W})$ in order to choose the best values for $\mathbf{W}$. To do this, a so-called validation dataset is constructed in exactly the same way as the training dataset, but with different $[\mathbf{u},\mathbf{v}]$ pairs (Prechelt, 1998a). This can then be used to calculate the so-called validation error for given $\mathbf{W}$, which is simply equation 2.11 evaluated for the validation dataset, rather than the training dataset. This can then be used to compare the performance of different values of $\mathbf{W}$ obtained from training runs with different initial values for $\mathbf{W}$ (or training parameters - see Appendices B and C).

Because the training dataset comprises a limited number of samples from the probability distribution $p(\mathbf{u},\mathbf{v})$, it may not *fully* sample the stochastic variation in $\mathbf{v}$. Therefore $\mathbf{q}(\mathbf{u};\mathbf{W})$ may fit the stochastic variation in $\mathbf{v}$ rather than reproducing the true variation in the expected value $\mathrm{E}[\mathbf{v}]$. Such over-fitting will reduce generalisation performance. It is usually observed that during gradient-descent the validation error initially decreases with iteration number, but after a certain number of iterations the validation error begins to increase (Prechelt, 1998b). This may be caused by training, having initially fitted the large-scale variation of the expected value, proceeding to fit small-scale stochastic variation. Thus using the validation error one may decide to terminate training before such 'noise' is fitted, a technique known as early-stopping, allowing us to retain $\mathbf{W}$ with best generalisation performance.

Furthermore, if we monitor the validation error with iteration (of gradient-ascent) then the choice of the number of hidden nodes is easy to make: we may simply choose an arbitrarily large number of hidden nodes (i.e., beyond what is deemed necessary by the number of training instances available to constrain the corresponding number of weights - see Bishop (1995, pp.128-132)) and use early-stopping to prevent over-fitting. Indeed, it is often observed that choosing a seemingly excessive number of hidden nodes aids generalisation by effectively introducing smoothing to the neural network output (Wang et al., 1994; Sarle, 1995).

2.7.4 Deep neural networks

The use of so-called deep neural neural networks, which are networks with more than one hidden layer of nodes, can improve generalisation performance greatly (Hinton and Salakhutdinov, 2006; Bengio et al., 2013). The fundamental advantage of deep

neural networks is that they can emulate complex mappings with fewer weights, since having numerous hidden layers permits a series of non-linear transformations to be applied to the input. Thus potentially less training data is required to train these networks (Håstad and Goldmann, 1991; Bengio and Delalleau, 2011). For a given training dataset, it is often noted that deep networks have better generalisation in classification (Ranzato et al., 2007) and regression problems (Parviainen, 2010) than networks with one hidden layer (but with equal numbers of weights, i.e., free-parameters, in the networks).

Furthermore, deep neural networks are often defined with a so-called bottle-neck layer, which is a layer with fewer nodes, or dimensions, than the input. This layer then represents a lower dimensional representation of the input. Thus, after training, it is hoped that this lower dimensional representation contains the most important features of the input for predicting the output, and any noise or superfluous information in the input will be removed (van der Maaten et al., 2009). Thus bottle-neck layers are useful for promoting good generalisation and reducing over-fitting, albeit at the risk of losing information by reducing the dimensionality of the input (Erhan et al., 2010). Such networks have recently been used successfully in a geophysical context to reduce the dimensionality of seismological data by Valentine and Trampert (2012).

Unfortunately, unlike networks with one hidden layer, deep neural networks are difficult to train and as such have not been used extensively in practice until recently (Bengio, 2012). The complex structure of deep networks means that naively using back-propagation often results in convergence in the weights' values to a local minima in equation 2.11 which yields a poor approximation of the desired mapping (Hochreiter, 1998). However, it has been shown recently that performing so-called pre-training (or conditioning) to determine initial values for $\mathbf{W}$, before attempting to minimise equation 2.11, can yield estimates for $\mathbf{W}$ which perform well (i.e., $\mathbf{q}(\mathbf{u};\mathbf{W})$ performs the mapping accurately and exhibits good generalisation). There are numerous existing pre-training approaches (see e.g., Hinton et al., 2006; Vincent et al., 2010; Valentine and Trampert, 2012), but here we use the so-called stacked denoising-autoencoder method (described in Appendix C), which has been shown to be effective for regression problems (Parviainen, 2010).

2.8 Application to a real dataset

We now apply our new method of elastic inversion to real data for the Laggan gas field, located in the West of Shetland hydrocarbon province. The reservoir is in Paleocene strata and consists of spatially extensive turbidite lobes. There are three such lobes each approximately 3km wide, 6km long and 10m thick. These lobes form homogeneous fine-grained sandstones which are referred to as the 'A', 'B' and 'C' sands. The sand units are separated by shale units, themselves thought to originate at the fringe of turbidite lobes or as channel levees (Gordon et al., 2010).

From the results of a seismic survey, AVA-type data had been produced at each lateral position in a 500×500 grid over the reservoir. Thus we had 2.5×10^5 traces of real AVA-type data: $\mathbf{d}^r_{\mathbf{x}}$, where we now use $\mathbf{x} = [x, y]$ to differentiate between traces at different lateral positions and $x \in [1, ..., 500]$ and $y \in [1, ..., 500]$. For each trace of real AVA-type data $Z = 150$, thus we defined a 3-D subsurface grid with $X = 500$, $Y = 500$ and $Z = 150$. The units of the vertical dimension z were vertical travel time, with each cell spanning $1ms$, and the lateral dimensions x and y were spatial, with each cell spanning $10m$. $Z = 150$ for all traces (of both real and synthetic quantities) used henceforth in this application of the method (however, for purposes of presentation, some figures will crop the extremities of these traces). We also obtained well data and a geological interpretation of the reservoir. Three wells (wells 1-3) intersect the reservoir, all of which have vertical or near-vertical well-bore trajectories. The outline of the reservoir, the data grid (i.e., all $\mathbf{x}$ positions) and position of the wells is shown in Figure 2.4.

Thus our aim was to obtain a (single) neural network mapping $\mathbf{q}$ which could be applied (via the recursive 1-D operation) to the results of deterministic elastic inversion at each lateral position where AVA-type data existed (over the Laggan field), i.e., $\forall\ \mathbf{x}$. To do this we followed the general 10-step workflow as outlined in section 2.4:

(1) Define the low-fidelity prior: The same Gaussian low-fidelity prior $p_L(\mathbf{e})$ (equation 2.1) was used for both the deterministic inversion of each $\mathbf{d}^r_{\mathbf{x}}$ for all $\mathbf{x}$ (step 2), and of the synthetic AVA-type data (step 6). Thus $\mathbf{e}_0$ was formed by applying a low frequency filter (a Butterworth filter with cut-off frequency $\omega_C = 25Hz$ and slope parameter $n = 4$) to measurements of $\mathbf{e}$ made at well 3 to obtain a general, regional depth trend. $\mathbf{\Sigma}_e$ was derived from a variogram calculated from the same

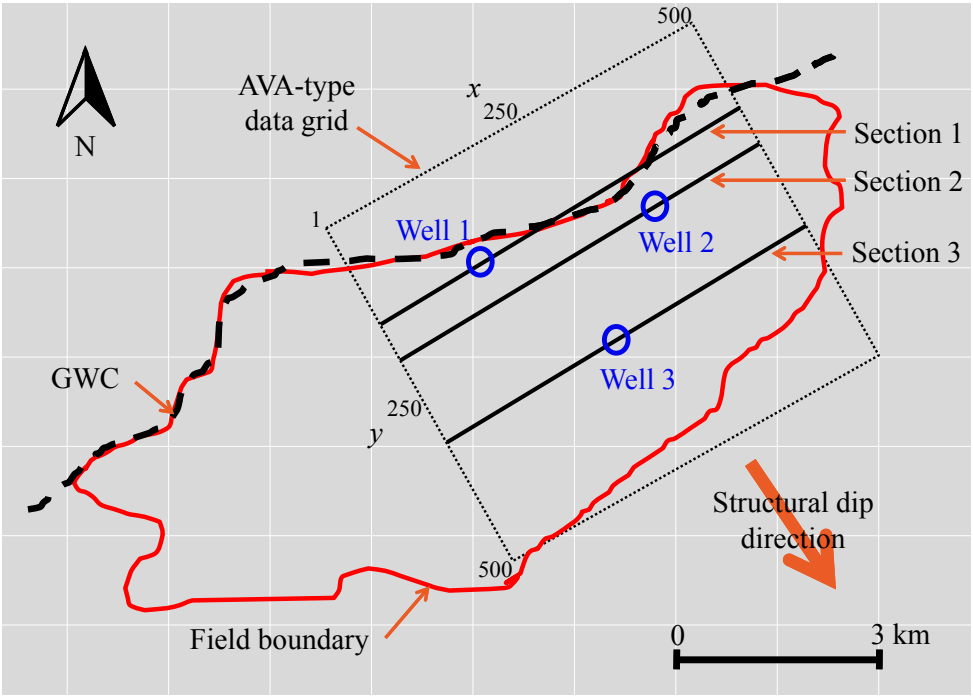

Figure 2.4: A map showing the outline of the Laggan gas field (red line). The gas-water contact (GWC) is shown at the base of the reservoir (with respect to its structural dip) by a black stippled line. The extent of the grid of AVA-type data traces, $\mathbf{d}^r_{\mathbf{x}}$, where $\mathbf{x} = [x, y]$ and $x \in [1, ..., 500]$ and $y \in [1, ..., 500]$, is shown by a black-dotted line. Cross-sections 1-3 are shown intersecting wells 1-3, respectively.

well measurements of $\mathbf{e}$.

(2) Deterministic inversion of the real AVA-type data: Estimates $\hat{\mathbf{e}}^r_{\mathbf{x}}\ \forall\ \mathbf{x}$ were obtained by solving (by gradient-descent) equation 2.2 using $\mathbf{d}^r_{\mathbf{x}}\ \forall\ \mathbf{x}$ and $p_L(\mathbf{e})\ \forall\ \mathbf{x}$ (from step 1). The matrices $\mathbf{\Sigma}_d$ and $\mathbf{S}$ (required to calculate $\mathbf{f}(\hat{\mathbf{e}})$ and hence solve equation 2.2) were determined from well data, and were assumed constant for all lateral positions $\mathbf{x}$ (and are retained for use in steps 4 and 5).

(3) Define the high-fidelity prior: To define $p_H(\mathbf{e})$ we used the geological interpretation of the reservoir to build a 1-D model. We did not expect to be able to resolve the thin 'A' and 'B' sands so we assumed these, and the separating shale layer, to be a single unit. Thus our model comprised: two sand layers (an 'A+B' and a 'C' layer), one separating shale layer, and an overburden and basal shale layer. We generated $B = 3000$ trace realisations of this 1-D facies model, where the thicknesses of the layers varied according to a normal distribution, whose mean and covariance was determined from the layer thicknesses measured down the wells.

(4) Generate synthetic elastic realisations: Histograms describing the probability of the I_P, I_S and ρ values in each facies were determined from the well-log data (Figure 2.6). For each of the B trace realisations of the 1-D facies model, these were sampled from to generate B traces of I_P, I_S and ρ. Thus we obtained $\mathbf{e}^s_b \sim p_H(\mathbf{e}),\ b \in [1, ..., B]$. It should be noted that correlation was introduced within both the layers' thicknesses (step 3) and the elastic parameters with respect to the b index. This allows us to form visually understandable 'cross-sections' of the synthetic facies and elastic parameters (Figures 2.5 and 2.7, respectively) by plotting the traces with respect to b. However, this is for aesthetic effect only: this 2-D lateral-correlation information cannot be encapsulated, or applied, by the 1-D recursive operation ($\mathbf{q}$) in step 10.

(5) Generate synthetic AVA-type data: Calculate $\mathbf{d}^s_b = \mathbf{f}(\mathbf{e}^s_b),\ b \in [1, ..., B]$, using $\mathbf{S}$ and $\mathbf{\Sigma}_d$ from step 1 in equation 1.6. The resulting cross-sections of 'near', 'mid' and 'far' angle-stack data are shown in Figure 2.8.

(6) Deterministic inversion of synthetic AVA-type data: Solve equation 2.2 for all $\mathbf{d}^s_b,\ b \in [1, ..., B]$, where for all b the same $p_L(\mathbf{e})$ PDF (from step 1) is used. The minimisation (i.e., gradient-descent) requires evaluation of $\mathbf{d} = \mathbf{f}(\hat{\mathbf{e}})$ using $\mathbf{S}$ and $\mathbf{\Sigma}_d$ from step 1 in equation 1.6. The results of inversion of each trace $\hat{\mathbf{e}}^s_b,\ b \in [1, ..., B]$ are shown in Figure 2.9.

(7) Define the training dataset: From steps 3 and 5 we have the set of pairs of

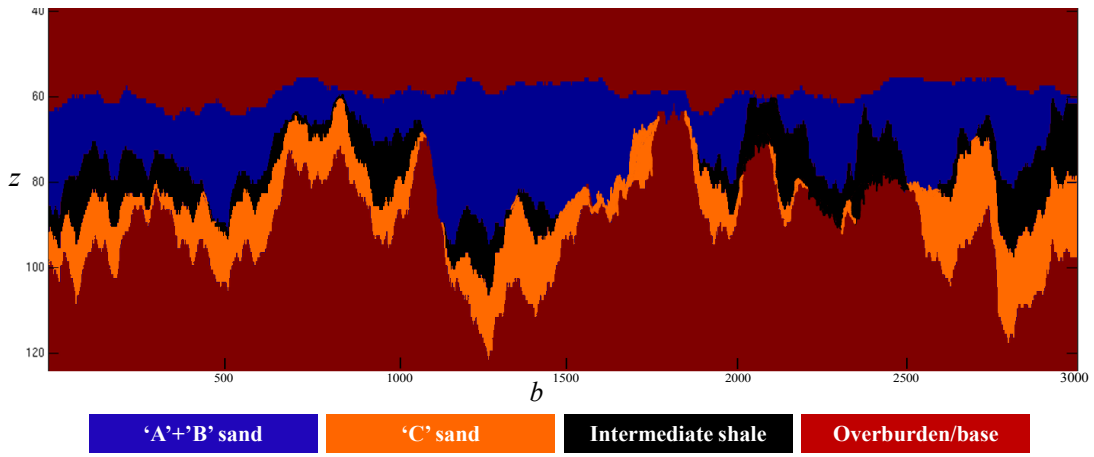

Figure 2.5: The set of $B = 3000$ 1-D facies models used to generate the training dataset for $\mathbf{q}$ plotted with respect to b. Each trace is generated by realising the 'A'+'B' sand layer, a separating shale layer, the 'C' sand layer and the overburden and basal shale layer. The thicknesses of the reservoir layers are realised from a normal distribution whose parameters are determined from well data. In order to generate an understandable geological image (i.e., a 2-D cross-section), correlation in layer thicknesses has been enforced with respect to b. However, the recursive operation (application of $\mathbf{q}$) cannot learn, or apply, this lateral correlation. Note that the traces in this image have been cropped to permit magnification of the reservoir layers.

vectors (traces) $[\hat{\mathbf{e}}_b^s, \mathbf{e}_b^s]$, $b \in [1, ..., B]$. The training dataset is generated by applying $\mathcal{Q}$ to each of the $B = 3000$ pairs in this set. λ is set equal to half the period of the longest wavelet ('far') in $\mathbf{S}$ which was $42ms$, hence $\lambda = 21$. Substituting λ and $Z = 150$ into equation 2.8 gives the number of training pairs which can be extracted from the single pair of vectors $[\hat{\mathbf{e}}_b^s, \mathbf{e}_b^s]$. Thus multiplying this by B gives the total number N of input-output pairs available to form the training dataset: $[\mathbf{u}_i^s, \mathbf{v}_i^s]$, $i = 1, 2, ..., N$ where $N = 3000 \times (150 - 42) = 324,000$.

(8) Define neural network q: The dimension of the input and output layers are equal to those of $\mathbf{u}$ and $\mathbf{v}$, respectively. Substituting λ into equation 2.7, the dimensions of $\mathbf{u}$ and $\mathbf{v}$ are $(43 + 21) \times 3 = 192$ and 3, respectively. We specified the topology of $\mathbf{q}$ to be deep and to have a bottleneck layer, with the topology expressed in short-hand being $193 - 500 - 100 - 500 - 3$ (including biases).

(9) Learn the neural network q: After applying pre-training (Appendix C) to obtain an initial $\mathbf{W}$ matrix, 200 iterations of gradient-descent were applied to minimise equation 2.11 with respect to $\mathbf{W}$. 100 such training runs were made (each with a different random seed). A validation dataset was generated (in the same way as the training dataset above but with $B = 200$), and thus the validation error

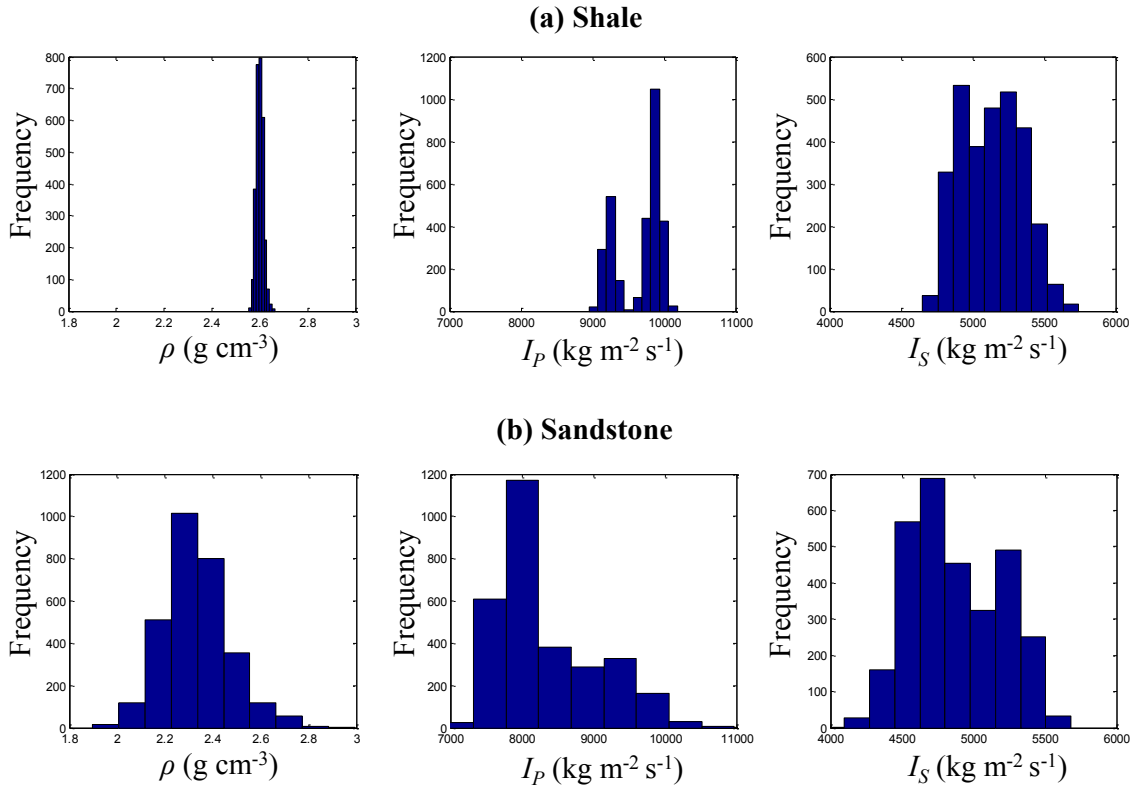

Figure 2.6: Histograms showing the distribution of the elastic parameters I_P, I_S and ρ in (a) the shale layers (basal, separating and overburden units), and (b) in the sandstone layers ('A', 'B' and 'C' units).

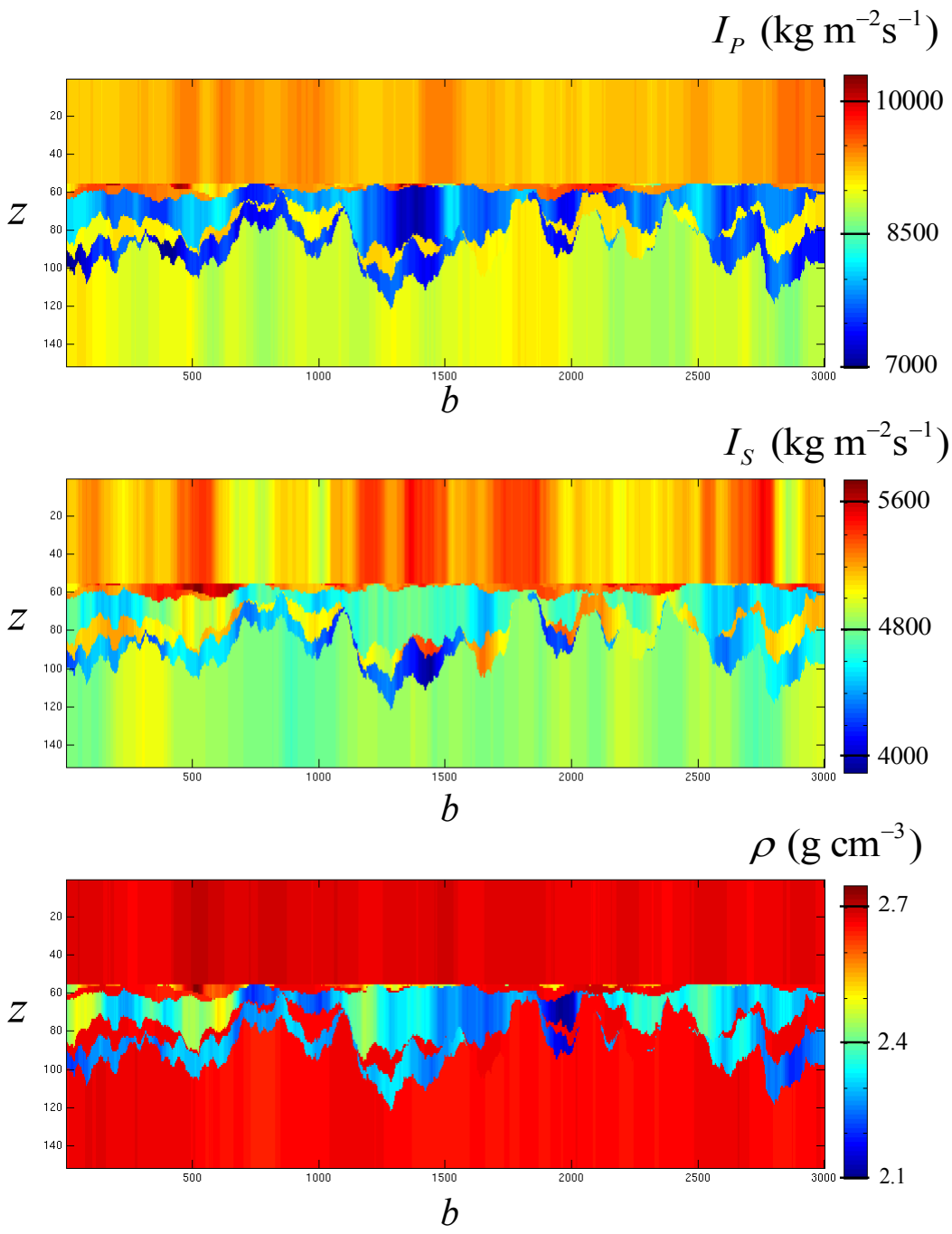

Figure 2.7: The set of $B = 3000$ 1-D synthetic elastic parameter models $\mathbf{e}_b^s$, $b \in [1, ..., B]$. This data forms part of both the output and input portion of the training dataset for $\mathbf{q}$. It was generated by populating each of the layers in each of the 1D facies models (Figure 2.5) with the elastic parameters, sampled from histograms derived from well data (Figure 2.6). In order to generate an understandable geological image (i.e., a 2-D cross-section), correlation in the elastic parameter values has been enforced with respect to b. However, the recursive operation (application of $\mathbf{q}$) cannot learn, or apply, this lateral correlation.

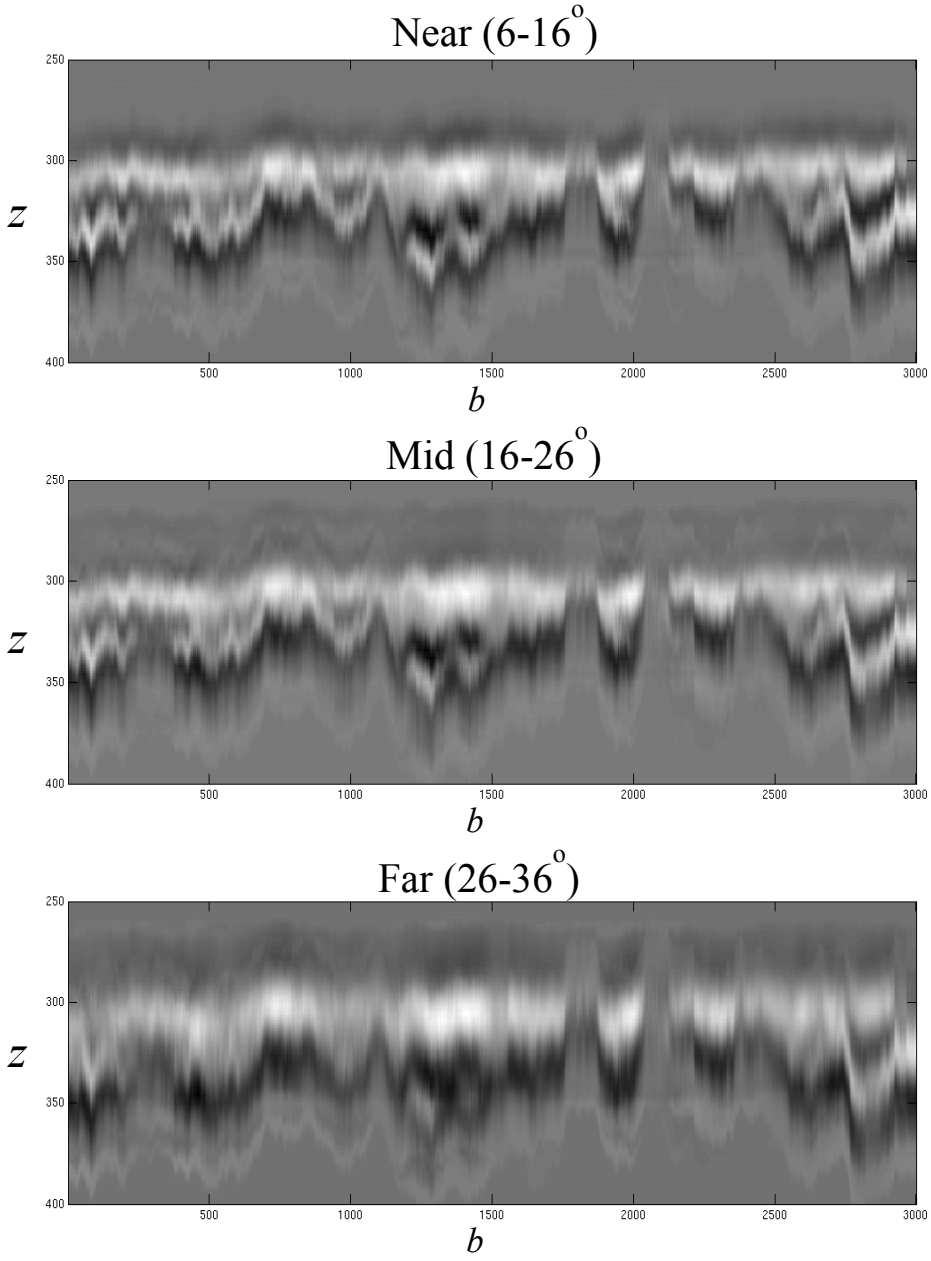

Figure 2.8: The synthetic AVA-type data generate by applying the AVA forward function (equation 1.6) to each of the 1-D elastic parameter model traces in Figure 2.7, i.e., $\mathbf{d}_b^s = \mathbf{f}(\mathbf{e}_b^s),\ b \in [1, ..., B]$, where $\mathbf{d}_b^s$ comprises 'near' $(6-16°)$, 'mid' $(16-26°)$ and 'far' $(26-36°)$ angle-stack data. This data is used to determine the synthetic deterministic inversion results which form part of the training dataset for $\mathbf{q}$. Note that the gray-scale represents normalised amplitude in each section.

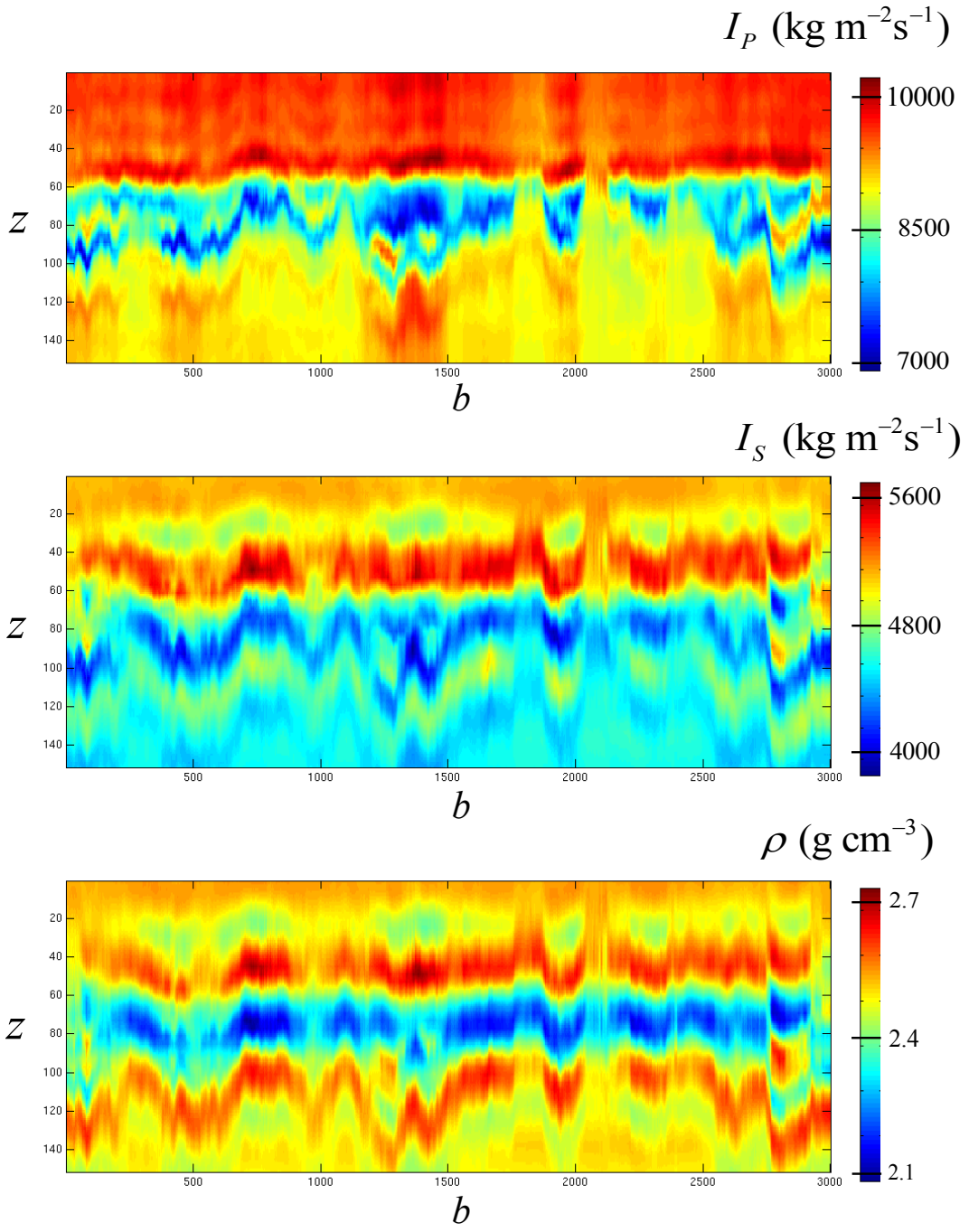

Figure 2.9: The results of deterministic seismic inversion applied to the synthetic AVA-type data (Figure 2.8), $\hat{\mathbf{e}}_b^s$, $b \in [1, ..., B]$. These results are used to form part of the input portion of the training dataset for $\mathbf{q}$.

was calculated at each iteration of gradient-descent, in each training run. Thus we retained the so-called optimal weights $\hat{\mathbf{W}}$ which was the $\mathbf{W}$ matrix which gave the lowest validation error of all iterations of all 100 training runs. A visual comparison of the output portion of the validation dataset and the output of $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$, supplied with the input portion of the validation dataset, is shown in Figure 2.10.

(10) Apply the recursive operation using q: $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ trained in step 6 was applied (via the 1-D recursive operation) to $\hat{\mathbf{e}}^r_{\mathbf{x}} \; \forall \; \mathbf{x}$. The resulting 'high-resolution' elastic parameter estimates are shown for three 2-D cross-sections from the survey. For comparison, the sections are chosen such that each coincides with a well trajectory down which $\mathbf{e}_{\mathbf{x}}$ has been measured: sections 1-3 are intersected by wells 1-3, respectively. The sections are shown in Figures 2.11-2.13, and Figures 2.14-2.16 show a magnified comparison of the well-measured, deterministic inversion estimates and neural network derived estimates of $\mathbf{e}_{\mathbf{x}}$ at each well position.

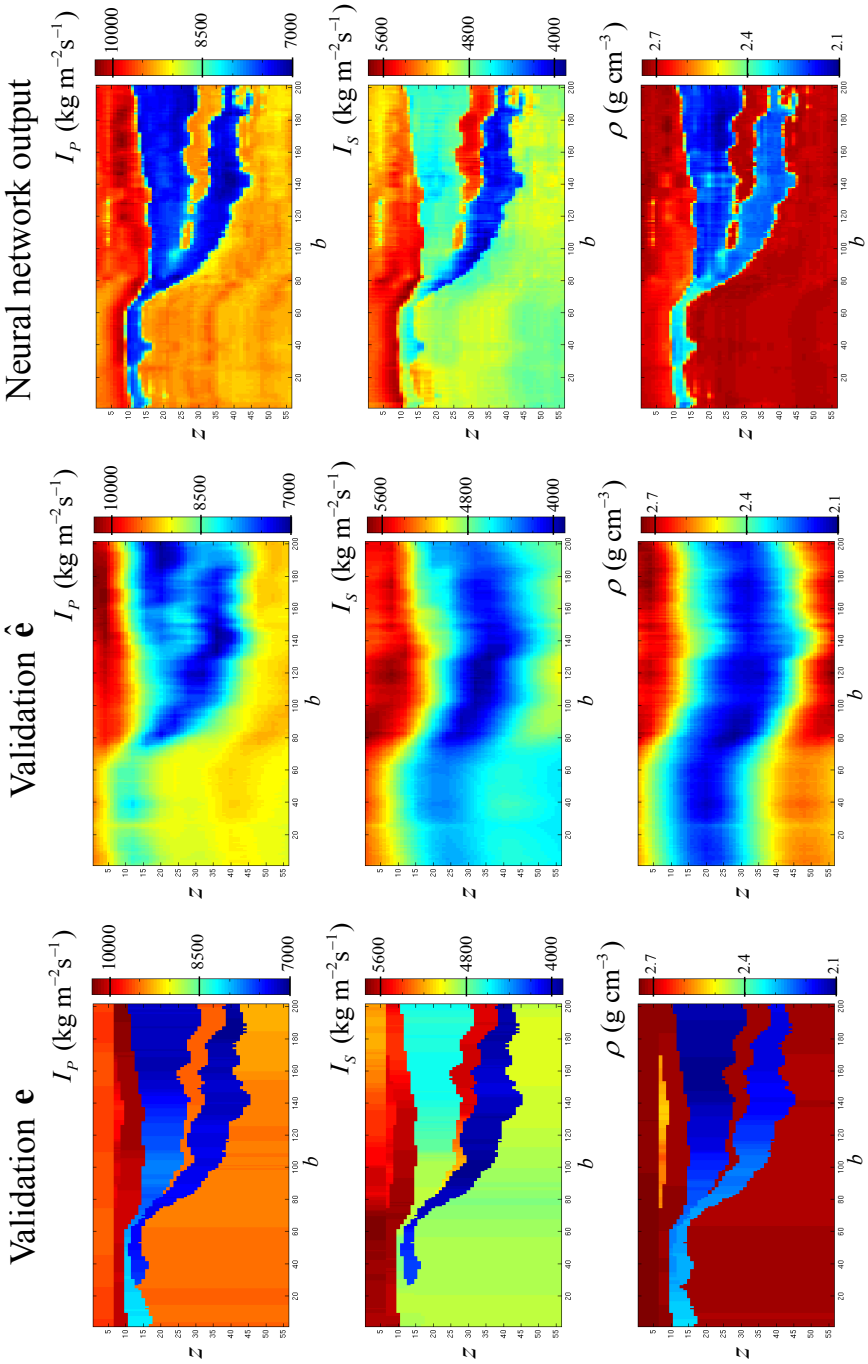

Figure 2.10: Results of applying the $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ (where $\hat{\mathbf{W}}$ are the optimal weights found by training) to the validation dataset. Left column: the validation elastic parameter $\mathbf{e}$ model which is used to generate the validation AVA-type data $\mathbf{d}$. Middle column: The results of deterministic inversion $\hat{\mathbf{e}}$ of the validation $\mathbf{d}$. Right column: the results of applying $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ to the validation $\hat{\mathbf{e}}$. Note that the validation error in this case would be calculated as the sum-of-square-errors between the results obtained with $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ (right column) and the 'true' elastic parameters (left column). Note that the traces in these images have been cropped to permit magnification of the reservoir layers.

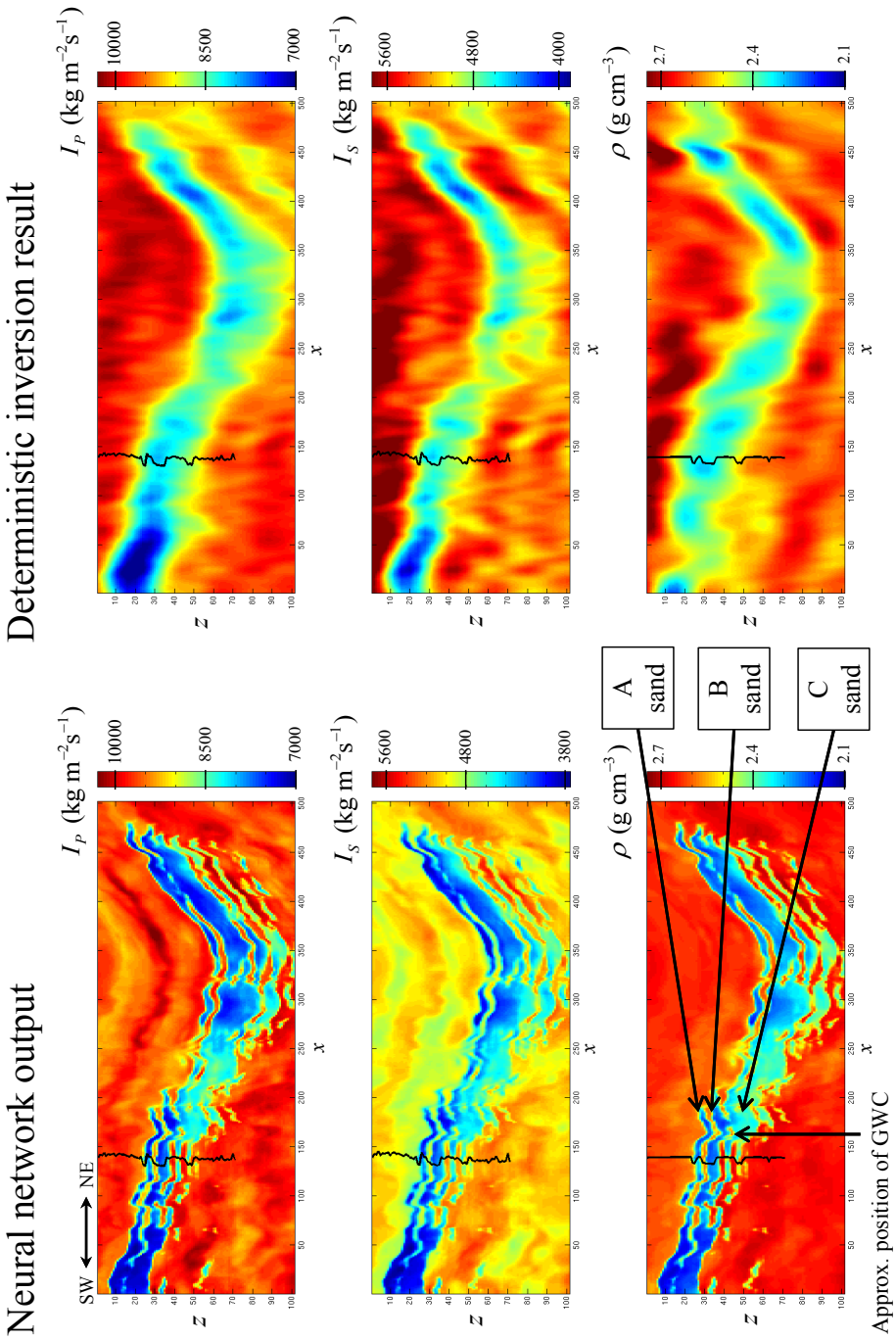

Figure 2.11: Results of applying $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ recursively (left column) to the results of deterministic inversion $\hat{\mathbf{e}}^r$ (right column) of real AVA-type data $\mathbf{d}^r$ for the Laggan field. The elastic parameters I_P, I_S and ρ are shown in each row. The position of this cross-section is marked on the base map (Figure 2.4) as section 1, and is intersected by well 1. The elastic parameters $\mathbf{e}$ measured down its (vertical) trajectory by well-logging are plotted on the appropriate cross-sections in black, for the purposes of identifying the true reservoir layer positions. Note that the traces in this image have been cropped to permit magnification of the reservoir layers.

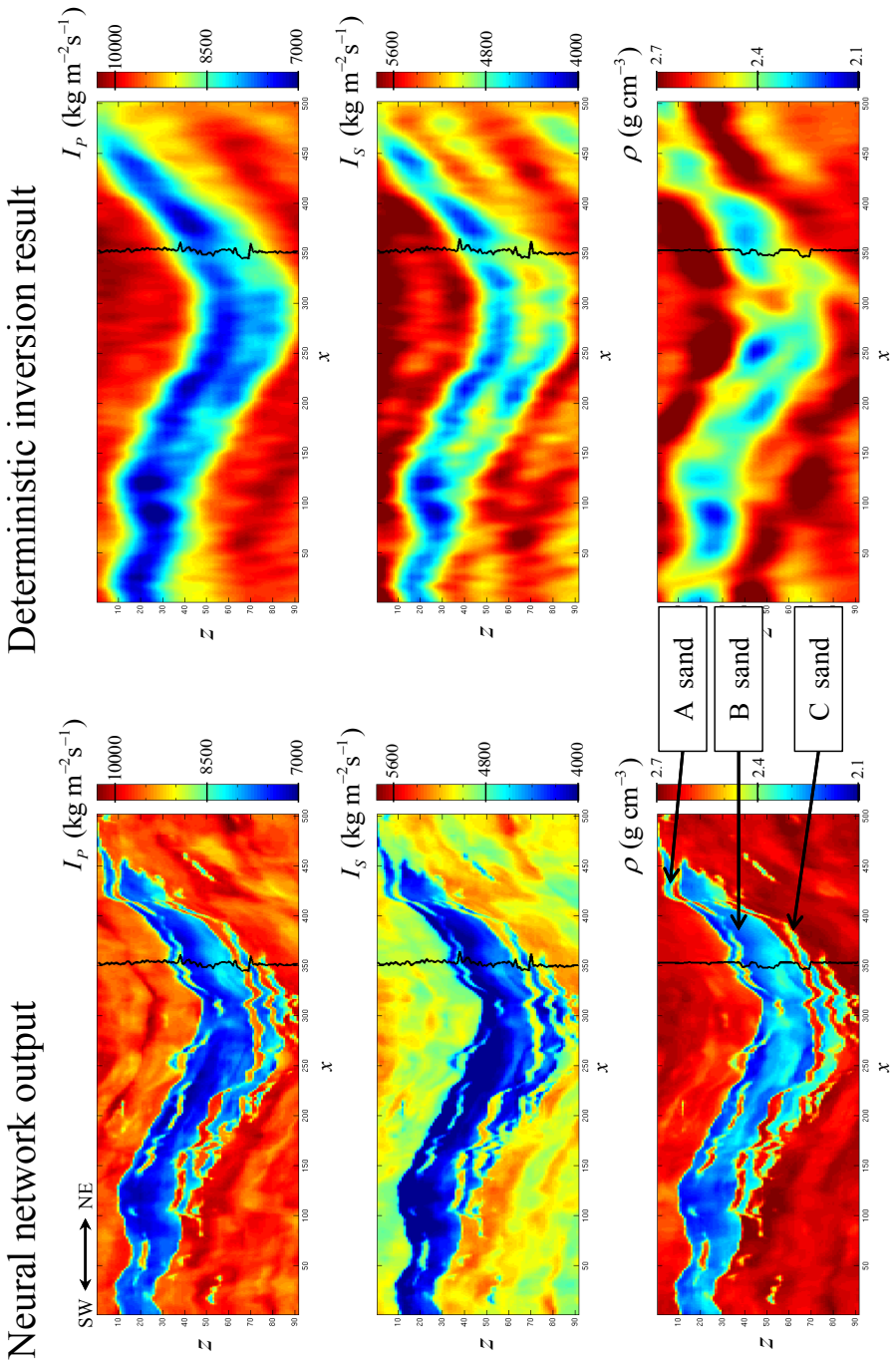

Figure 2.12: As for Figure 2.11 but for section 2 intersected by well 2.

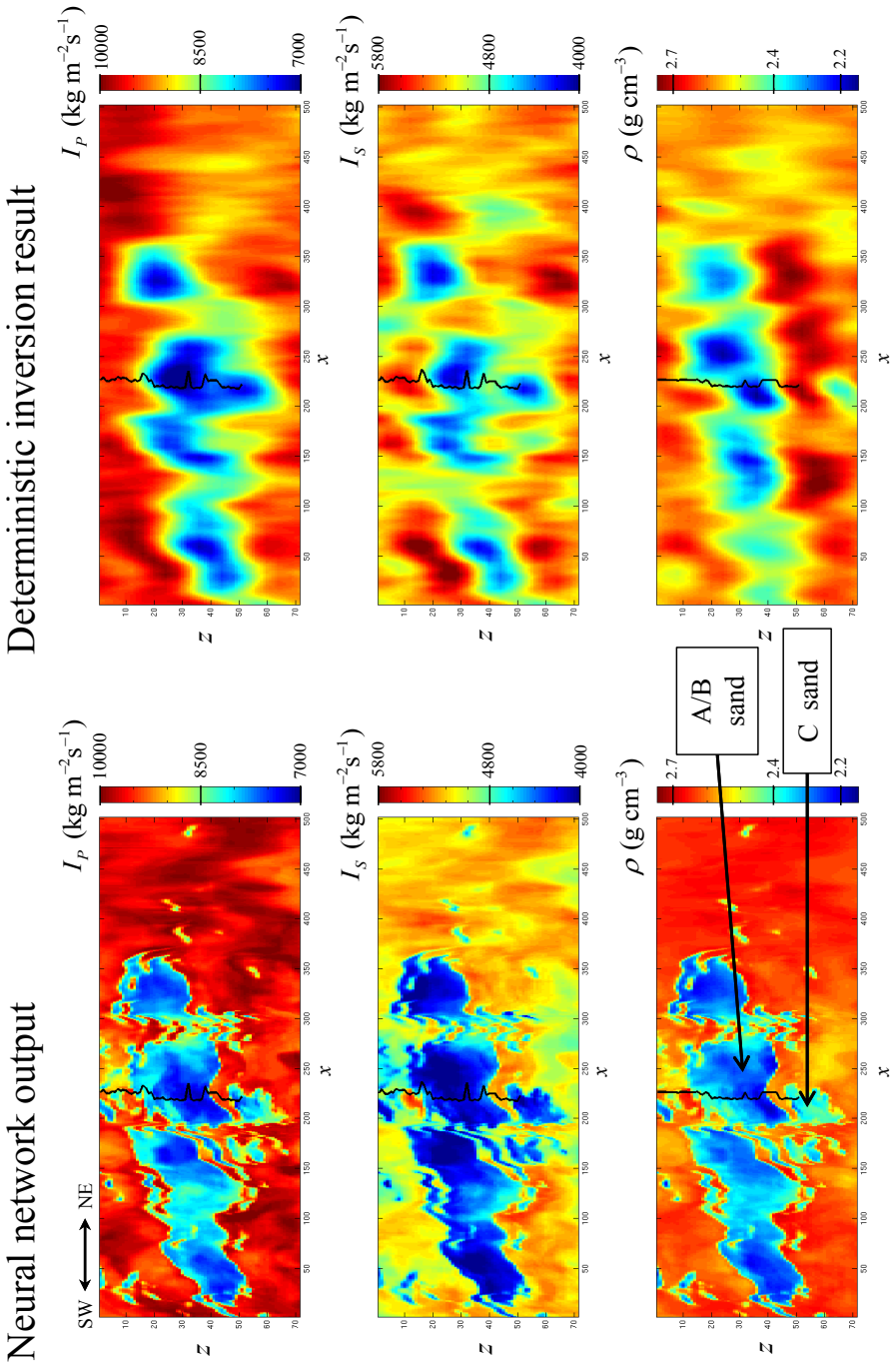

Figure 2.13: As for Figure 2.11 but for section 3 intersected by well 3.

2.9 Discussion

In the training procedure above each of the 100 training runs took $\sim$ 1800 seconds. A large number of training runs was required since $\sim$ 90% of training runs (even with the autoencoder pre-training method) yielded values for $\mathbf{W}$ which produced poor results when $\mathbf{q}(\mathbf{u};\mathbf{W})$ was applied to the validation dataset (i.e., the sections in the right column of Figure 2.10 would be geologically unreasonable in these cases). Additionally, a considerable amount of time was spent trialling different network topologies (numbers of hidden layers and nodes) and training parameters (ϕ and η, defined in Appendix B and C) in order to find a configuration which yielded the best training results. However, once trained the recursive application of $\mathbf{q}(\mathbf{u};\hat{\mathbf{W}})$ to all time samples down a trace at position $\mathbf{x}$, and hence estimation of $\mathbf{e}_{\mathbf{x}}$, was very rapid taking < 1 second (for each trace).

The results obtained by the recursive application of the optimal neural network $\mathbf{q}(\mathbf{u};\hat{\mathbf{W}})$ are geologically reasonable in all of the cross-sections (Figures 2.11-2.13). The sand and shale layers, as well as the effects of faulting, are better resolved in comparison to the results of deterministic inversion. Furthermore, we find that the results of the deep neural network methodology closely match the well-log measurements at well 1 (Figure 2.14). However, at wells 2 and 3 (Figures 2.15 and 2.16) the positioning of the layers and their estimated elastic parameter values are inaccurate. Interestingly, the neural network results have been able to resolve all three sandstone layers ('A', 'B' and 'C') individually in some locations (e.g., Figure 2.14) even though the 1-D facies model used to generate the training dataset (Figure 2.5) only contained two sandstone layers ('A+B' and 'C'); the neural network has learnt to 'recognise', in a general sense, a layer in the input data irrespective of position in the trace. Such 'position invariance' is only possible because $\mathbf{q}$ is defined to act recursively.

The poor results at wells 2 and 3 may be caused by inaccuracy in the deterministic inversion results $\hat{\mathbf{e}}^r$, which form the neural network's input. The deterministic inversion methodology used does not constrain $\hat{\mathbf{e}}^r$ to fit the elastic parameter values measured at the wells. The well data is only used to specify the low-frequency model $\mathbf{e}_0$ and the covariance matrix $\mathbf{\Sigma}_{\mathbf{e}}$. Thus $\hat{\mathbf{e}}^r$ may be as inaccurate at the well positions as anywhere else in the survey. We can assess the accuracy of deterministic inversion results at the well positions by reducing the band-width of the well-log data to match that of $\hat{\mathbf{e}}^r$. To this end, we applied a low-pass filter (a Butterworth filter

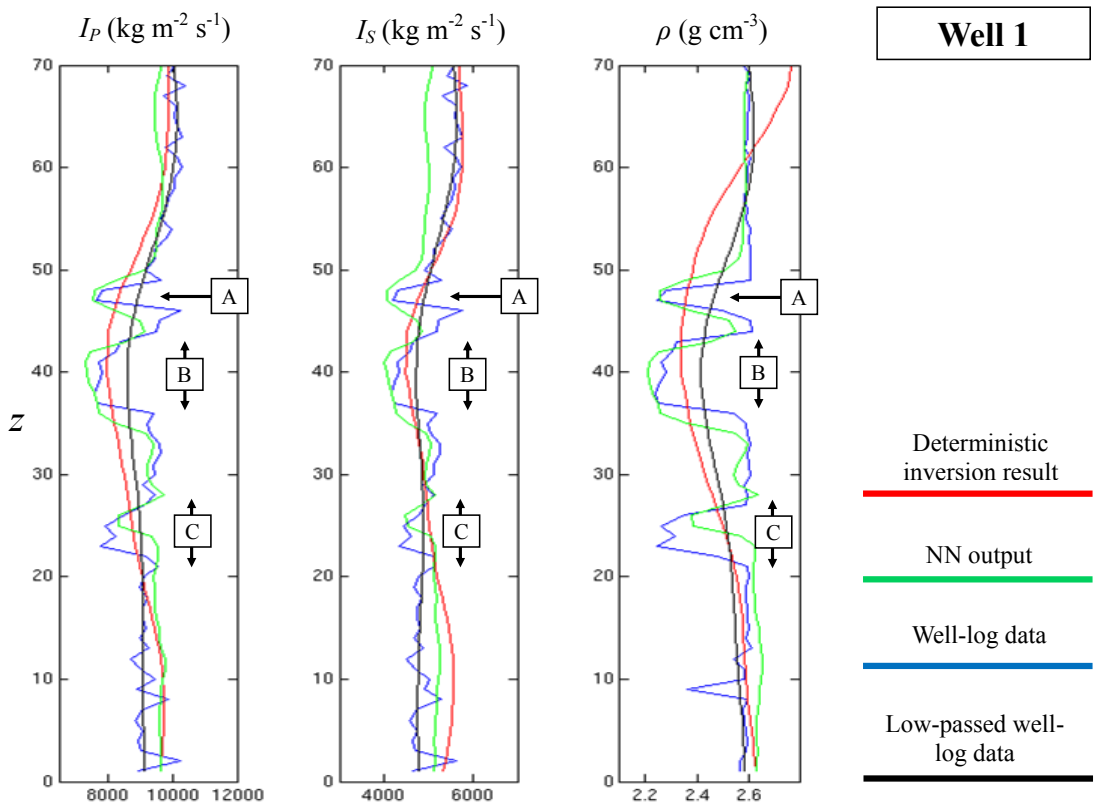

Figure 2.14: Results of applying $\mathbf{q}(\mathbf{u}; \hat{\mathbf{W}})$ recursively (green line) to the results of deterministic inversion $\hat{\mathbf{e}}^r$ (red lines) down the vertical trajectory of well 1 (see Figure 2.4). The elastic parameters $\mathbf{e}$ measured by well-logging (blue lines) are shown for comparison to the neural network results. These measurements were also band-limited (using a low-pass filter) to match the band-width of the deterministic inversion results $\hat{\mathbf{e}}^r$ for comparison (black lines). The I_P, I_S and ρ elastic parameters are shown in the left, middle and right columns, respectively. The approximate positions of the 'A', 'B' and 'C' sandstone layers are marked. Well 1 intersects section 1 and thus its lateral position in x is plotted in Figure 2.11. Note that these traces have been cropped to permit magnification of the reservoir layers.

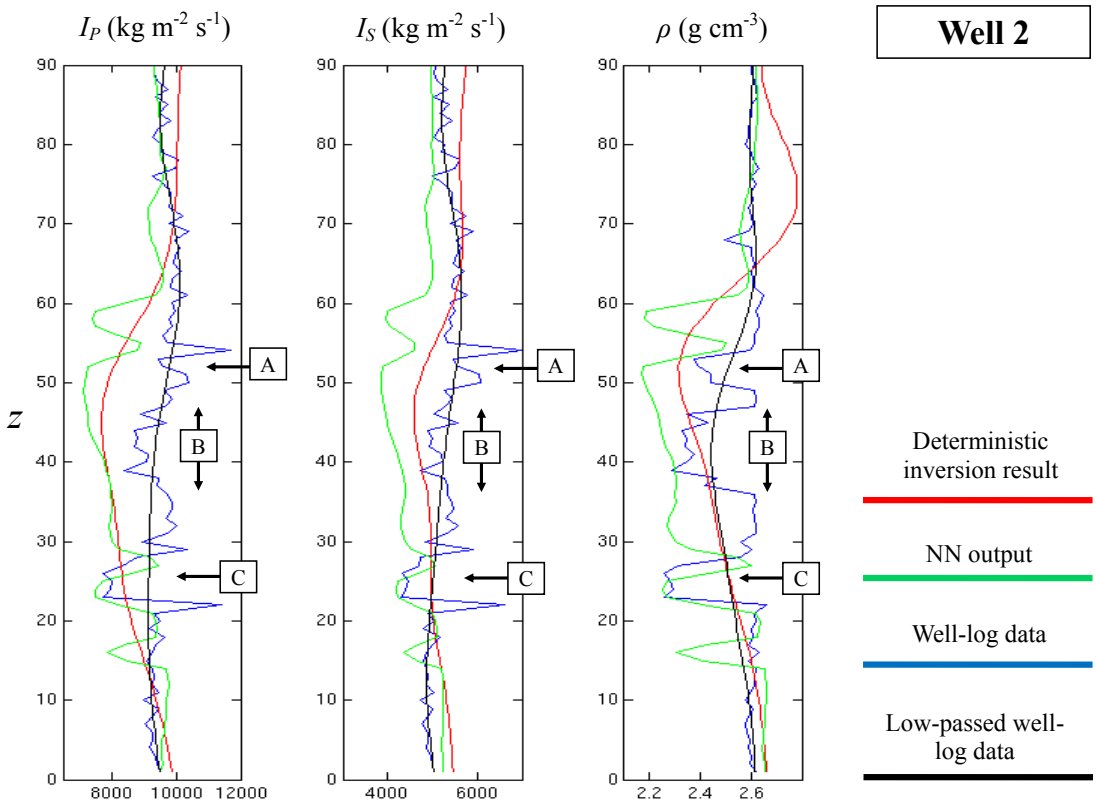

Figure 2.15: As for Figure 2.14 but for well 2, which intersects section 2 (Figure 2.12).

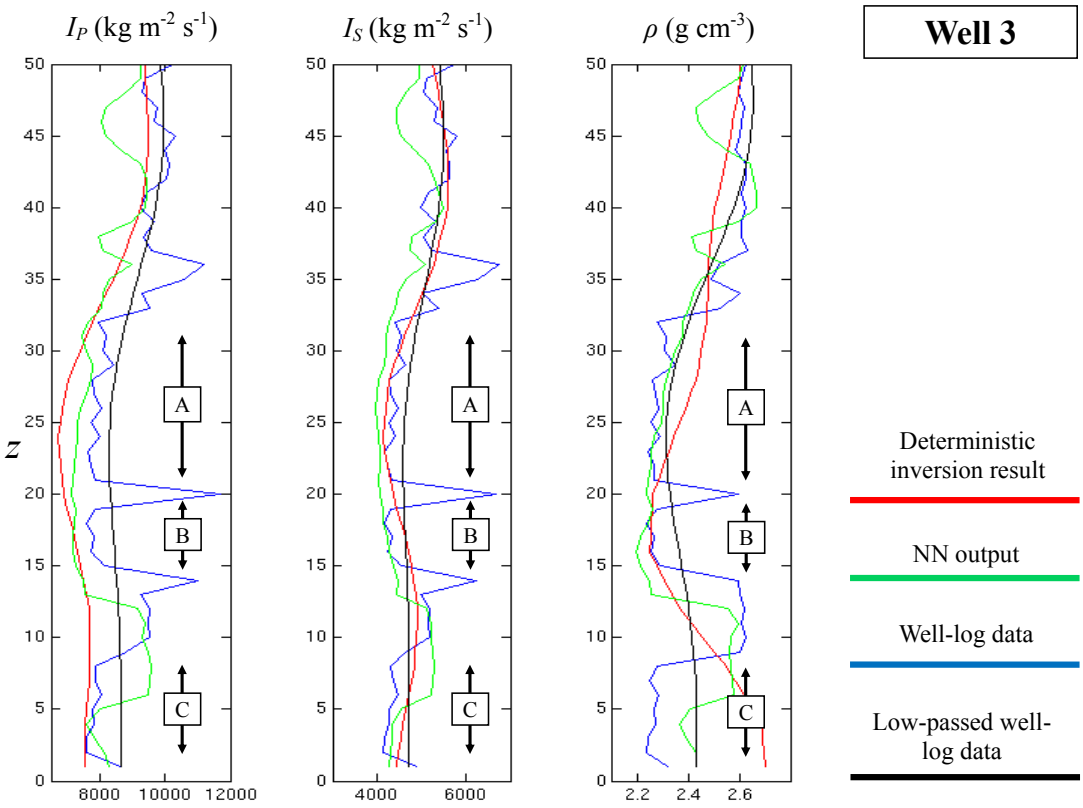

Figure 2.16: As for Figure 2.14 but for well 3, which intersects section 3 (Figure 2.13).

with cut-off frequency $\omega_C = 100Hz$ and slope parameter $n = 4$) to the well measurements of $\mathbf{e}$. The resulting reduced bandwidth well measurements (black lines) are plotted alongside $\hat{\mathbf{e}}^r$ (red lines) in Figures 2.14-2.16. From this, it is clear that the results of deterministic inversion are very inaccurate at the positions of wells 2 and 3, compared to those at well 1. This mirrors the performance of the neural network for estimating $\mathbf{e}$ and suggests that inaccuracy in the deterministic inversion results could be a significant cause of inaccuracy in the neural network's predictions.

Random noise in the AVA-type data is an obvious source of inaccuracy in the deterministic inversion results, especially for the low and high frequency ranges which have the lowest signal-to-noise ratio. Additionally, processing errors in the AVA-type data and lateral variation in the source wavelets (such that our assumption of invariant wavelets in equation 1.6 breaks down) may cause inaccuracy in $\hat{\mathbf{e}}^r$. In fact, the deterministic inversion algorithm which we employed implemented lateral correlation (i.e., with $\mathbf{x}$) as a constraint on the solution $\hat{\mathbf{e}}^r$, which should reduce the effect of these errors (Thore, 2013). Furthermore, it ensures a geologically reasonable solution in terms of continuity of $\hat{\mathbf{e}}^r$ between traces. Although lateral correlation is not considered in the formulation of our methodology (since $\mathbf{q}$ is only applied via the 1-D recursive operation), this lateral correlation in the input data is clearly influential in ensuring that the final results which we obtain using our method exhibit lateral continuity (i.e., with respect to $\mathbf{x}$, and are hence geologically reasonable).

It is clear that in future we must improve the accuracy of the deterministic inversion stage. However, it would be simpler and more efficient to avoid deterministic inversion altogether: we could train $\mathbf{q}$ to *directly* take the AVA-type data $\mathbf{d}^s$ as input and output an estimate of $\mathbf{e}$ (containing the high-fidelity prior information). We could use the same recursive neural network function but train it with the training dataset $[\mathbf{d}_i^s, \mathbf{e}_i^s],\ i \in [1, ..., N]$. However, lateral continuity, and hence compensation for error in the AVA-type data, would no longer be applied by deterministic inversion. This is the most likely reason that we have, as yet, been unable to successfully apply the current 1-D method directly to real AVA-type data.

To overcome this problem we might define $\mathbf{q}$ as a 3-D recursive operator, which is applied sequentially through a 3-D volume of AVA-type data. This would permit both lateral continuity to be enforced and 3-D multi-point prior geological information to be implemented (in the same way that the current methodology implements 1-D multi-point information down each trace). However even for the 1-D mapping, deep neural network training is computationally costly and requires a large amount

of user interaction, as demonstrated above. A neural network which takes a 3-D segment of data as input would require many more input variables within the network topology, and consequently many more weights to accommodate the increased complexity of the mapping. Furthermore, a larger training dataset would be required in order to constrain these extra parameters. Thus the computational and user-time cost of training a 3-D $\mathbf{q}$ may be very high, and improved (possibly automated) (pre-)training procedures are required to make training such networks possible in practice. Such advances may ultimately lead to a methodology which permits the full non-approximated *probabilistic* mapping (equation 2.5 in the 1-D case) to be learnt by a deep neural network.

Despite the limitation of our method to the 1-D transformation of the results of deterministic inversion, it can nevertheless be of practical use for applying sophisticated prior information and hence for obtaining high resolution estimates of subsurface elastic parameters. This was demonstrated by the results obtained for the Laggan dataset (but only when *accurate* deterministic inversion results have been obtained). It should be noted that the neural network and deterministic inversion results are not directly comparable; we have attempted to apply as much prior information as possible using the recursive neural network, whereas we have specified that the deterministic inversion use only very restricted prior information (since we use a Gaussian low-fidelity prior). We have not attempted to compare our methodology to other methods which attempt to constrain seismic inversion using high-fidelity prior information (e.g., González et al., 2007), which may be an interesting topic for future study.

2.10 Summary

A new reservoir characterisation methodology was introduced which transforms the results of deterministic elastic inversion for subsurface elastic parameters to estimates of those parameters with high vertical resolution. The methodology uses a neural network to approximately emulate the mapping between the deterministic elastic inversion results and the 'true' elastic parameters. The neural network function is designed to be applied recursively down a trace of deterministic estimates, predicting as output the 'true' elastic parameters and taking as input (a portion of) the deterministic elastic estimates and the previously predicted 'true' elastic parameters

down the trace. The neural network acts only on traces at a single lateral location in isolation thus the method is strictly one-dimensional.

The parameters of the neural network which emulate the desired mapping are obtained via a training process, using a set of example instances of the input and output variables. This training dataset is generated by first making samples from a probability distribution which accurately encapsulates prior knowledge of the subsurface elastic parameters, to obtain examples of the output. Then, to obtain samples of the input, synthetic seismic data is generated from these, and this seismic data is inverted deterministically to create corresponding examples of deterministic inversion estimates. To ensure that the neural network learns the mapping robustly, a deep topology was chosen for the network. This promoted good generalisation of the trained neural network to inputs which were dissimilar to those in the training dataset.

The new methodology was tested on a real dataset for the Laggan gas field. A geological interpretation of the reservoir and well data was used to generate one-dimensional realisations of the elastic parameters in the reservoir. This accurate prior information, and the corresponding results of deterministic inversion, formed the training dataset. After training, the deep neural network was applied to real deterministic elastic inversion results for the reservoir, yielding estimates of the elastic parameters with greatly increased vertical resolution. It was found that these results were consistent with measurements of the elastic parameters at well positions where the results of deterministic inversion were accurate. Thus it was shown that the deep neural network methodology is useful for improving deterministic elastic inversion results by applying sophisticated (multi-point geostatistical) prior information, which was one of the aims set in section 1.8.

References

Barber, D., and C. M. Bishop (1998), Ensemble learning in Bayesian neural networks, *NATO ASI Series F Computer and Systems Sciences*, *168*, 215–238.

Bengio, Y. (2012), Practical recommendations for gradient-based training of deep architectures, in *Neural Networks: Tricks of the Trade,* pp.437–478, Springer.

Bengio, Y., and O. Delalleau (2011), On the expressive power of deep architectures, in *Algorithmic Learning Theory,* pp.18–36, Springer.

Bengio, Y., A. Courville, and P. Vincent (2013), Representation learning: A review and new perspectives, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *35*(8), 1798–1828.

Bishop, C. M. (1994), *Mixture density networks, Tech. Rep.NCRG/94/0041*, Dept. of Computer Science and Applied Mathematics, Aston University.

Bishop, C. M. (1995), *Neural networks for pattern recognition*, Oxford University Press.

Caers, J. (2001), Geostatistical reservoir modelling using statistical pattern recognition, *Journal of Petroleum Science and Engineering*, *29*(3), 177–188.

Erhan, D., Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio (2010), Why does unsupervised pre-training help deep learning?, *The Journal of Machine Learning Research*, *11*, 625–660.

González, E. F., T. Mukerji, and G. Mavko (2007), Seismic inversion combining rock physics and multiple-point geostatistics, *Geophysics*, *73*(1), R11–R21.

Gordon, A., T. Younis, C. Bernard-Graille, R. Gray, J.-M. Urruty, L. Ben-Brahim, J.-C. Navarre, B. Paternoster, and G. Evers (2010), Laggan; a mature understanding of an undeveloped discovery, more than 20 years old, in *Petroleum Geology Conference series,* pp.279–297, Geological Society of London.

Gubbins, D. (2004), *Time series analysis and inverse theory for geophysicists*, Cambridge University Press.

Håstad, J., and M. Goldmann (1991), On the power of small-depth threshold circuits, *Computational Complexity*, *1*(2), 113–129.

Hinton, G. E., and R. R. Salakhutdinov (2006), Reducing the dimensionality of data with neural networks, *Science*, *313*(5786), 504–507.

Hinton, G. E., S. Osindero, and Y.-W. Teh (2006), A fast learning algorithm for deep belief nets, *Neural computation*, *18*(7), 1527–1554.

Hochreiter, S. (1998), The vanishing gradient problem during learning recurrent neural nets and problem solutions, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *6*(02), 107–116.

Parviainen, E. (2010), Dimension reduction for regression with bottleneck neural networks, in *Intelligent Data Engineering and Automated Learning–IDEAL 2010,* pp.37–44, Springer.

Prechelt, L. (1998a), Automatic early stopping using cross validation: Quantifying the criteria, *Neural Networks*, *11*(4), 761–767.

Prechelt, L. (1998b), Early stopping - but when?, in *Neural Networks: Tricks of the trade*pp.55–69, Springer.

Ranzato, M., F. J. Huang, Y.-L. Boureau, and Y. Lecun (2007), Unsupervised learning of invariant feature hierarchies with applications to object recognition, in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07.,* pp.1–8, IEEE.

Sarle, W. S. (1995), Interface Foundation of North America.

Shahraeeni, M. S., A. Curtis, and G. Chao (2012), Fast probabilistic petrophysical mapping of reservoirs from 3D seismic data, *Geophysics*, *77*(3), O1–O19.

Thore, P. (2013), Data driven versus model based inversion - When and why?, in *75th EAGE Conference Exhibition extended abstracts*, EAGE.

Valentine, A. P., and J. Trampert (2012), Data space reduction, quality assessment and searching of seismograms: Autoencoder networks for waveform data, *Geophysical Journal International*, *189*(2), 1183–1202.

van der Maaten, L. J., E. O. Postma, and H. J. van den Herik (2009), Dimensionality reduction: A comparative review, *Journal of Machine Learning Research*, *10*(1-41), 66–71.

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol (2010), Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *The Journal of Machine Learning Research*, *9999*, 3371–3408.

Wang, C., S. S. Venkatesh, and J. S. Judd (1994), Optimal stopping and effective machine complexity in learning, *Advances in neural information processing systems*, *6*, 303–310.

Chapter 3

Prior replacement for geological inversion

3.1 Overview

In section 1.7 we described how neural network methods can be used to efficiently solve the geological inverse problem by determining, and repeatedly applying, the mapping $\hat{\mathbf{e}}_i \rightarrow p(\mathbf{g}_i|\hat{\mathbf{e}}_i) \ \forall \ i$. However, the mapping is determined only for a given prior distribution $p(g_i)$ implicit in the cell-wise geological posterior $p(\mathbf{g}_i|\hat{\mathbf{e}}_i)$. Thus, if such neural network methods are to be more useful generally for geological inversion, then a method for varying this prior with i, which does not require the costly re-training of the neural network, is needed. Therefore in this chapter we introduce an operation which subverts the usual order of application of Bayes' rule in Bayesian inversion: we take a probability already created using Bayes' rule, and remove the prior probability, replacing it with a different prior probability. We call this operation *prior replacement*.

3.2 Introduction

Later in this chapter we will demonstrate prior replacement for continuous geological parameters. Thus it is convenient to derive prior replacement using m_i as the model parameter (instead of g_i) in this chapter. However, the prior replacement operation can be applied equally to discrete and continuous model parameters. For continuous parameters the cell-wise geological posterior distribution is written using Bayes' rule

as

$$p(\mathbf{m}_i|\hat{\mathbf{e}}_i) = \frac{p(\hat{\mathbf{e}}_i|\mathbf{m}_i)p(\mathbf{m}_i)}{p(\hat{\mathbf{e}}_i)}, \tag{3.1}$$

where all distributions are probability density functions (PDFs), and as usual in Bayes' rule $p(\mathbf{m}_i)$ is the prior distribution and $p(\hat{\mathbf{e}}_i|\mathbf{m}_i)$ is the likelihood distribution (that is, the cell-wise geological likelihood as described in section 1.3). $p(\hat{\mathbf{e}}_i) = \int_{-\infty}^{+\infty} p(\mathbf{e}_i|\mathbf{m}_i)p(\mathbf{m}_i)d\mathbf{m}_i$ is the normalising constant (where we now use the integration limits $\int_{-\infty}^{+\infty}$ since $\mathbf{m}_i$ is continuous).

Using Bayes' rule we can now see that in principle prior replacement is a simple calculation: roughly speaking, we divide the posterior distribution in equation 3.1 by the existing prior, $p(\mathbf{m}_i)$ and multiply by the new prior distribution. Thus we *replace* the prior in equation 3.1 with the new prior. We have only found two explicit treatments of this operation in the literature, both in reference to statistical classification models - that is, probabilistic classification of objects into discrete classes based on associated data (Michie et al., 1994). Bishop (1995, p. 223) uses prior replacement to modify the outputs of a Bayesian classification neural network, and Bailer-Jones and Smith (2010) use the term 'prior replacement' to describe the operation for discrete classification problems. However, neither work discusses how it may be applied to continuous model parameters, nor any potential uses for the operation in a wider context.

In this chapter we first describe the prior replacement operation in detail in section 3.4. Following this, we describe how neural networks are currently used to perform geological inversion for continuous model parameters (i.e., how the mapping $\mathbf{m}_i \rightarrow p(\mathbf{m}_i|\hat{\mathbf{e}}_i)$ is obtained) using so-called mixture density neural network (MDN) inversion in section 3.5. Then we describe how prior replacement may be applied to its results in section 3.6, in order to permit variation of $p(g_i)$ with i. In section 3.7 we give a numerical example of the application of prior replacement to the results of MDN inversion for the inversion of elastic parameters for geological parameters at a single grid cell i (section 3.7.1), and to a reservoir-scale geological inversion, where $p(g_i)$ varies for all i within a 2-D subsurface model grid (section 3.7.2).

Finally, we discuss the implications of our results with respect to both seismic inversion and Bayesian inversion in general. We also discuss the effect of prior replacement on the quality of the final posterior estimate obtained. The discussion of quality is supported by results presented in Appendix F for a simple Bayesian inverse problem example. These results also suggest that prior replacement may be

used as a variance reduction technique similar to importance sampling (indeed, prior replacement seems to outperform importance sampling in this respect for the simple problem presented therein).

3.3 Notation

The notation used in this chapter follows that used in Chapter 1 except for the use of continuous (i.e., $\mathbf{m}_i$), instead of discrete (i.e., g_i) geological parameters, for the description of geological inversion. Also, an example of prior replacement within geological inversion will be given where density is not considered in the elastic parameter vector, i.e., $\hat{\mathbf{e}}_i = [I_P, I_S]_i$ here. Note that, as per the workflow illustrated in Figure 1.1, it is assumed that $\hat{\mathbf{e}}_i$ represents the results of deterministic elastic inversion in this chapter. However, in order to simplify notation we discontinue the use of the hat symbol, and use $\mathbf{e}_i$ in place of $\hat{\mathbf{e}}_i$ in this chapter. A summary of the notation used in this chapter is given in Appendix H.2.

3.4 Probabilistic development of prior replacement

To derive the prior replacement operation in general terms we now write out the Bayesian solution to an inverse problem in two different situations. Both situations involve an inverse problem with the same forward function, thus the likelihood distribution is identical in both. However, in the first, so-called 'old' situation there is a different prior probability distribution to that of the second 'new' situation. We denote these with 'old' and 'new' subscripts. It follows from Bayes' theorem that the posterior must also vary. Accordingly the normalising constant may also vary, which can be seen if we write it in the integral form in the denominator of Bayes' theorem for the two situations:

$$p_{old}(\mathbf{m}_i|\mathbf{e}_i) = \frac{p(\mathbf{e}_i|\mathbf{m}_i)p_{old}(\mathbf{m}_i)}{p_{old}(\mathbf{e}_i)} = \frac{p(\mathbf{e}_i|\mathbf{m}_i)p_{old}(\mathbf{m}_i)}{\int_{-\infty}^{+\infty} p(\mathbf{e}_i|\mathbf{m}_i)p_{old}(\mathbf{m}_i)d\mathbf{m}_i} \tag{3.2}$$

and

$$p_{new}(\mathbf{m}_i|\mathbf{e}_i) = \frac{p(\mathbf{e}_i|\mathbf{m}_i)p_{new}(\mathbf{m}_i)}{p_{new}(\mathbf{e}_i)} = \frac{p(\mathbf{e}_i|\mathbf{m}_i)p_{new}(\mathbf{m}_i)}{\int_{-\infty}^{+\infty} p(\mathbf{e}_i|\mathbf{m}_i)p_{new}(\mathbf{m}_i)d\mathbf{m}_i}. \tag{3.3}$$

We can therefore see that $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ can be written in terms of $p_{old}(\mathbf{m}_i|\mathbf{e}_i)$ (and vice versa) by

$$p_{new}(\mathbf{m}_i|\mathbf{e}_i) = p_{old}(\mathbf{m}_i|\mathbf{e}_i)\frac{p_{new}(\mathbf{m}_i)}{p_{old}(\mathbf{m}_i)}\frac{p_{old}(\mathbf{e}_i)}{p_{new}(\mathbf{e}_i)}. \tag{3.4}$$

In the context of inversion, we are usually supplied with a fixed data vector $\mathbf{e}_i$. Hence, in both new and old situations we assume that the data observed is the same. The normalising constant is dependent upon the form of the prior so may vary between the two situations. Nevertheless it is still independent of the value of the parameter vector $\mathbf{m}_i$. Therefore, for later convenience we set $p_{new}(\mathbf{e}_i)/p_{old}(\mathbf{e}_i) = k$, such that

$$p_{new}(\mathbf{m}_i|\mathbf{e}_i) = \frac{1}{k}\frac{p_{new}(\mathbf{m}_i)}{p_{old}(\mathbf{m}_i)}p_{old}(\mathbf{m}_i|\mathbf{e}_i). \tag{3.5}$$

Equation 3.5 now has a form which allows us to evaluate the new posterior distribution from the old one, assuming that we know both the old and the new prior, $p_{old}(\mathbf{m}_i)$ and $p_{new}(\mathbf{m}_i)$ respectively, and that we can evaluate the scale factor k. The latter can be shown to be a normalising constant: since from the definition of a PDF we have that $\int_{-\infty}^{+\infty} p_{new}(\mathbf{m}_i|\mathbf{e}_i)d\mathbf{m}_i = 1$, so integrating over both sides of equation 3.5 yields

$$k = \int_{-\infty}^{+\infty} \frac{p_{new}(\mathbf{m}_i)}{p_{old}(\mathbf{m}_i)}p_{old}(\mathbf{m}_i|\mathbf{e}_i)d\mathbf{m}_i. \tag{3.6}$$

Equation 3.5 shows the main operation involved in prior replacement. It will yield a valid result only under certain conditions. One can interpret equation 3.5 as trying to correct for a prior that is incorrect. The old posterior is divided by the old prior in an attempt to remove its effects. If the old prior had regions of zero probability then this will result in undefined values (0/0) where the old prior and posterior are simultaneously zero in the model space. We can interpret this as follows: when the old prior was initially applied and the old posterior obtained, we lost all information about the likelihood in those regions, and we cannot regain such information by changing the prior. Thus we are forced to assume that these undefined regions still have zero probability if we wish to continue. We implement this through our new prior: it is a condition that this must have zero probability where the old prior had zero probability, hence the new posterior will have zero probability in such areas too. We refer to this as the *support condition* henceforth.

3.5 Mixture density neural network inversion for geological inversion

In Chapter 2 we used a neural network to emulate the *deterministic* mapping between an estimate of the elastic parameters and an estimate of those parameters with improved vertical resolution. As discussed in that chapter it would have been preferable to obtain a probabilistic mapping, but the dimensionality of the data prohibited practical implementation of the appropriate neural network methodologies. In this case we are considering a problem that has a much lower dimensionality, since the sample spaces of $\mathbf{m}_i$ and $\mathbf{e}_i$ ($\mathcal{G}$ and $\mathcal{E}$, respectively) are relatively small. In this case it is feasible to obtain a neural network mapping which does the mapping from a datum to a probability distribution, that is to emulate $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i)$.

This can be achieved using mixture density network (MDN) inversion as used by Shahraeeni and Curtis (2011). MDN inversion is based on the assumption that any posterior PDF like that in equation 3.1 can be approximated by the sum of K normalised multivariate Gaussians each weighted by a constant (Bishop, 1994, 1995; McLachlan and Peel, 2004)

$$p(\mathbf{m}_i|\mathbf{e}_i) = \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right), \tag{3.7}$$

where $\{\alpha_j | j \in 1, 2, ..., K\}$ are normalising weights which obey $\sum_{j=1}^{K} \alpha_j = 1$, and $\phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)$ is a normalised multivariate Gaussian function of $\mathbf{m}_i$ with mean $\boldsymbol{\mu}_j$ and covariance $\boldsymbol{\Sigma}_j$ (where normalised implies that $\int_{-\infty}^{+\infty} \phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) d\mathbf{m}_i = 1$). This approximation of a PDF by a series of weighted, normalised Gaussians is referred to henceforth as a Gaussian mixture model (GMM). Note that in this chapter (and Appendix D) the letter K, which represents the number of kernels in the GMM, should not be confused with k, which is used to represent normalising constants.

In MDN inversion, a neural network (see section 2.7) is determined that can predict values of α_j, $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ in the mixture model which approximate the correct posterior (the left hand side of equation 3.7) for any given value of $\mathbf{e}_i$. The weights of the neural network with such properties are estimated by training using samples from the distribution $p(\mathbf{m}_i, \mathbf{e}_i) = p(\mathbf{m}_i)p(\mathbf{e}_i|\mathbf{m}_i)$. Samples in this training dataset are obtained by first sampling from the parameter space using the prior distribution $p(\mathbf{m}_i)$, then obtaining the corresponding samples of $\mathbf{e}_i$ from the probabilistic forward

function (i.e., the cell-wise geological likelihood distribution - see section 1.3) $p(\mathbf{e}_i|\mathbf{m}_i)$ which is known. In principle, the training process for a MDN is the same as that for a regular neural network (as described in section 2.7), but there are some differences due to the definition of the MDN output as the parameters of a mixture model. For a full description of MDN training see Bishop (1995, pp.140-161) for isotropic Gaussian kernels, or Shahraeeni and Curtis (2011) who extended the method to anisotropic Gaussian kernels.

It must be noted that the distribution $p(\mathbf{e}_i|\mathbf{m}_i)$ when used to generate training data is only assumed to be known as a function of $\mathbf{m}_i$ (that is to say it is a probabilistic *forward* model). Of course, if it were known as a function of $\mathbf{e}_i$ then inversion would not be required.

Once trained the neural network can determine the posterior $p(\mathbf{m}_i|\mathbf{e}_i)$ corresponding to any $\mathbf{e}_i$ vector extremely rapidly and efficiently (i.e., do the mapping $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i)$). However, a trained MDN embodies the prior distribution $p(\mathbf{m}_i)$ used to generate its training data, and thus application of the neural network to predict $p(\mathbf{m}_i|\mathbf{e}_i)$ is strictly valid only where that prior is deemed appropriate. As argued in section 1.7, this is highly inappropriate in the case of geological inversion, where we wish to use the MDN to calculate $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i)\ \forall\ i$, and $p(\mathbf{m}_i)$ will certainly vary with respect to i in a reservoir model grid. Of course, the neural network could be re-trained at each cell with a different, appropriate prior. We refer to this methodology as the *prior-specific training* method, since the MDN is trained for a specific prior distribution in each cell. However, training is a computationally costly procedure and may even require numerous training 'runs' in order to obtain a neural network which yields reasonable estimates of the posterior (as was the case in Chapter 2). The number of cells in a subsurface model grid (i.e., M) may be millions or even billions, thus we argue that the use of prior replacement to vary the prior in the results of MDN inversion, instead of prior-specific training, can lead to great efficiency gains, and in the next section we show how prior replacement may be applied to the results of MDN inversion.

3.6 Prior replacement in MDN inversion

We can directly apply the prior replacement equations 3.1 to 3.6 to the results of MDN inversion. If we equate the old posterior that appears in these equations to

the mixture model output of the MDN then

$$p_{old}(\mathbf{m}_i|\mathbf{e}_i) = \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right). \tag{3.8}$$

for some set of α_j. Substitution of equation 3.8 into equations 3.5 and 3.6 permits us to write

$$p_{new}(\mathbf{m}_i|\mathbf{e}_i) = \frac{1}{k}\frac{p_{new}(\mathbf{m}_i)}{p_{old}(\mathbf{m}_i)} \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) \tag{3.9}$$

and

$$k = \int_{-\infty}^{+\infty} \frac{p_{new}(\mathbf{m}_i)}{p_{old}(\mathbf{m}_i)} \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) d\mathbf{m}_i. \tag{3.10}$$

Thus, equations 3.9 and 3.10 provide a method of performing prior replacement for the output of a MDN (i.e., for a GMM). As with the general equations for prior replacement (equations 3.5 and 3.6), these equations only have well defined results for $p_{old}(\mathbf{m}_i)$ and $p_{new}(\mathbf{m}_i)$ distributions that satisfy the support condition. However, an added complication arises because in equations 3.9 and 3.10 we use a GMM approximation to the posterior, $p_{old}(\mathbf{m}_i|\mathbf{e}_i)$. This GMM approximation is non-zero everywhere (except in the impractical case of Gaussian kernels with zero variance); the real $p_{old}(\mathbf{m}_i|\mathbf{e}_i)$ may not be non-zero everywhere, hence the non-zero nature of the GMM is an artefact of the approximation. Therefore $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ should still be zero wherever $p_{old}(\mathbf{m}_i)$ is zero (from equation 3.5). Since we know that the GMM approximation is in error in this case, we should therefore still apply a new prior $p_{new}(\mathbf{m}_i)$ which has zero probability where the old prior has zero probability. In other words, the support condition still holds in this instance.

In Appendix D, the prior replacement operations for MDNs are developed in more detail for certain analytical forms of the priors (old and new). We show that if the new prior is Gaussian or Uniform, and the old prior is Uniform, that equations 3.9 and 3.10 can be written as truncated GMMs (we will later make use of these derivations). However due to this truncation they cannot be integrated analytically (Drezner, 1992), so numerical integration techniques must be used to determine the normalising constant. By contrast in Appendix D.5 we also show that, if both old and new prior distributions are Gaussian, equations 3.9 and 3.10 are themselves GMMs, and as such analytical integration can be used to solve them. We will not use these derivations in the following examples, but we include them since they potentially

permit the prior replacement operation to be performed extremely rapidly. They are also of interest mathematically since they involve the division of Gaussians: this operation is non-trivial compared to the multiplication of Gaussians, and is only possible under certain conditions on the old and new priors. Whilst Gaussian multiplication is widespread in the literature (Tarantola, 2002; Buland and Omre, 2003; Petersen and Pedersen, 2006), we have found little reference to such a 'Gaussian division' operation elsewhere.

3.7 Testing prior replacement in MDN inversion

We compared the accuracy and computational efficiency of prior-specific training to prior replacement for a synthetic geological inverse problem solved using MDN inversion. To do this we first defined an uncertain forward relationship $\mathbf{m}_i \rightarrow \mathbf{e}_i$ using the PDF $p(\mathbf{e}_i|\mathbf{m}_i)$. This is the equivalent of the cell-wise geological likelihood (discussed in section 1.3) for continuous geological parameters. To define this PDF we used a variant of a well-known rock physics model, the Yin-Marion shaley-sand model (Marion, 1990; Yin et al., 1993), which has been used previously as the forward model in MDN inversion (Shahraeeni, 2011, p.16).

We used this model to predict two elastic parameters: the S-wave (I_S) and P-wave (I_P) impedances, given two continuous geological parameters: the clay content by volume (m_1) and the sandstone matrix porosity (m_2) of a rock comprising a mixture of sandstone and shale. The Yin-Marion model is in principle a deterministic model, however Gaussian noise is added to its output, thus its output may be expressed using $p(\mathbf{e}_i|\mathbf{m}_i)$. A full description of how the Yin-Marion shaley-sand model is used to define this distribution is given in Appendix E. Note that here we have set the pore fluid of the rock to be pure water in each cell (which is to say that the water saturation parameter, defined in Appendix E, $m_3 = 1 \; \forall \; i$).

We assume that the impedances I_S and I_P have been estimated by deterministic elastic inversion. Thus we have an estimated elastic parameter vector $\mathbf{e}_i = [I_P, I_S]_i$ at each cell in a subsurface model grid; thus we construct an inverse problem for $\mathbf{m}_i = [m_1, m_2]_i$ to be solved at each of the M cells in the grid. In section 3.7.1 we perform the MDN inversion $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i)$ for a single datum $\mathbf{e}_i$ (i.e., at a single cell in the model) and vary the prior using both prior replacement and prior-specific training, which permits us to compare the accuracy of the cell-wise posterior estimate

returned by the two methods. In section 3.7.2 we then test MDN inversion with prior replacement on a reservoir-scale grid model.

3.7.1 Prior replacement compared to prior-specific training at a single cell

Before testing the two methods we must train an MDN. As explained in section 3.5, to do this the probabilistic forward function is used in conjunction with a prior to generate samples from $p(\mathbf{e}_i, \mathbf{m}_i)$ to form a training dataset. In prior-specific training, samples are made directly from the new prior. For prior replacement, sampling is initially made from a Uniform old prior $p_{old}(\mathbf{m}_i)$ which was chosen to be as broad as possible in the context of the model space, i.e.,

$$p_{old}(\mathbf{m}_i) = p_{old}(m_1, m_2) = \begin{cases} 0 & \text{for } m_j \notin [0,1],\ j = 1,2 \\ 1 & \text{otherwise} \end{cases} \tag{3.11}$$

This old prior is then replaced by the new prior in each case. Note that all physically possible (see definition of m_1 and m_2 in Appendix E) $p_{new}(\mathbf{m}_i)$ PDFs are contained within the bounds of the Uniform distribution in equation 3.11, thus the support condition will hold for any $p_{new}(\mathbf{m}_i)$ chosen.

We now test prior replacement and prior-specific training (for a single $\mathbf{e}_i$ vector) for the case of a (i) Uniform, and (ii) Gaussian new prior. The test uses an entirely synthetic inversion: the data inverted by the MDN was also generated using $p(\mathbf{e}_i|\mathbf{m}_i)$; the same data $\mathbf{e}_i$, was used in both cases. It was chosen arbitrarily, since we simply use it to demonstrate the method. In each of cases (i) and (ii) the appropriate prior replacement equations in Appendix D were solved. The particular procedures for each case are described below. In order to make the comparison fair between the results of prior replacement and prior-specific training, an equal number of kernels were used: $K = 20$ in equations 3.7 through 3.10 for all MDNs trained in the following examples. Since the data point was the same in both cases, the same old posterior PDF was used for prior replacement of the Uniform and Gaussian priors. This PDF is shown in Figure 3.1.

A Markov-chain Monte-Carlo (MCMC) solution was obtained for reference in each case. This PDF was generated by taking $> 10^4$ samples from the appropriate posterior and then estimating the densities using a GMM with a very large number

of kernels. Because a large number of samples were taken, we can effectively consider this as the true posterior PDF. This is supported by the fact that the magnitude of autocorrelation between samples within the Markov-chain, in both cases, was typically much less than 0.01 at lags greater than 15 samples. The time taken to make the samples from the posterior using MCMC, and the time taken in fitting the density to these, is far in excess of the time required by the MDNs to return a posterior estimate. However, we do not seek to compare the efficiency of MDN inversion to MCMC methods (the advantages in terms of efficiency have already been demonstrated by Shahraeeni et al. (2012) and references therein): we only use the MCMC results for a comparison of solution quality.

(i) Uniform new prior

In order to perform prior replacement in this instance, equations D.15 and D.14 were evaluated. Numerical integration techniques were used to calculate the normalising constant in equation D.15. Figure 3.2(a) shows the new Uniform prior (that is, the prior which we want to apply). Figure 3.2(b) shows the MCMC solution for $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$. Figure 3.2(c) shows the estimate of $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ obtained using prior-specific training of a MDN with the Uniform new prior. Figure 3.2(d) shows the estimate of $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ obtained by using prior replacement to replace the old prior implicit within the old posterior in Figure 3.1 by the Uniform new prior in Figure 3.2(a).

(ii) Gaussian new prior

In order to perform prior replacement in this case, equations D.19 through D.23 were evaluated. Numerical integration techniques were used to calculate the normalising constant in equation D.23. Figure 3.3(a) shows the new Gaussian prior (that is, the prior we wish to apply). Figure 3.3(b) shows the MCMC solution for $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$. Figure 3.3(c) shows the estimate of $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ obtained using prior-specific training of a MDN with the Gaussian new prior. Figure 3.3(d) shows the estimate of $p_{new}(\mathbf{m}_i|\mathbf{e}_i)$ obtained by using prior replacement to replace the old prior implicit within the old posterior in Figure 3.1 by the Gaussian new prior in Figure 3.3(a).

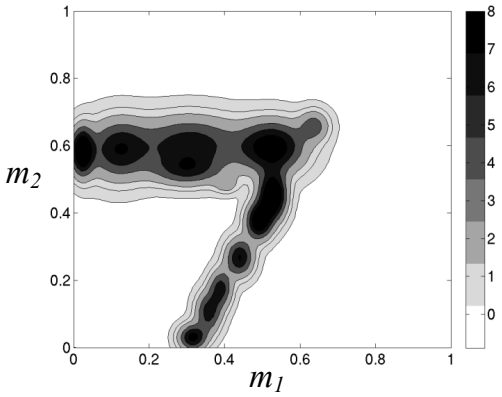

Figure 3.1: (a) The old posterior obtained from the output of a neural network (MDN) trained with samples made from the broad old prior defined in equation 3.11. Prior replacement was applied to this PDF to emplace a Uniform and a Gaussian new prior in Figures 3.2 and 3.3, respectively.

3.7.2 Application to reservoir-scale geological inversion

The results for the inversion of a single datum show that although variations exist, prior replacement and prior-specific training give comparable results; thus prior replacement is shown to work in practice using MDNs. Furthermore, we may conclude that the prior replacement method would always be faster than prior-specific training if many such inversions were performed, and if prior information varies between inversions. This can be understood by considering the computation times in the examples given: for prior replacement it took $\sim 10^2$ seconds to train the MDN using the old prior, then $\sim 10^{-3}$ seconds to run the MDN to obtain outputs for any given datum. Using prior replacement to construct the posterior for a new prior PDF for both the Uniform new prior (solving equations D.15 and D.14) and Gaussian new prior (solving equations D.23 and D.22) took $\sim 10^{-2}$ seconds. The total cost of prior replacement is therefore $\sim 10^2 + q \times (10^{-3} + 10^{-2})$ seconds, where q is the number of times prior information changes. For prior-specific training it also took $\sim 10^2$ seconds to train the MDN and again $\sim 10^{-3}$ seconds to run the MDN to obtain the outputs for a given datum. However, a new MDN has to be trained each time the prior changes so the total cost of prior-specific training is $\sim q \times (10^2 + 10^{-3})$ seconds. Therefore it is clear that if we were to apply both methods to the inversion of a large amount of data with varying priors then prior replacement could be orders of magnitude faster than prior-specific training.

This is the case when solving a geological inverse problem similar to the above,

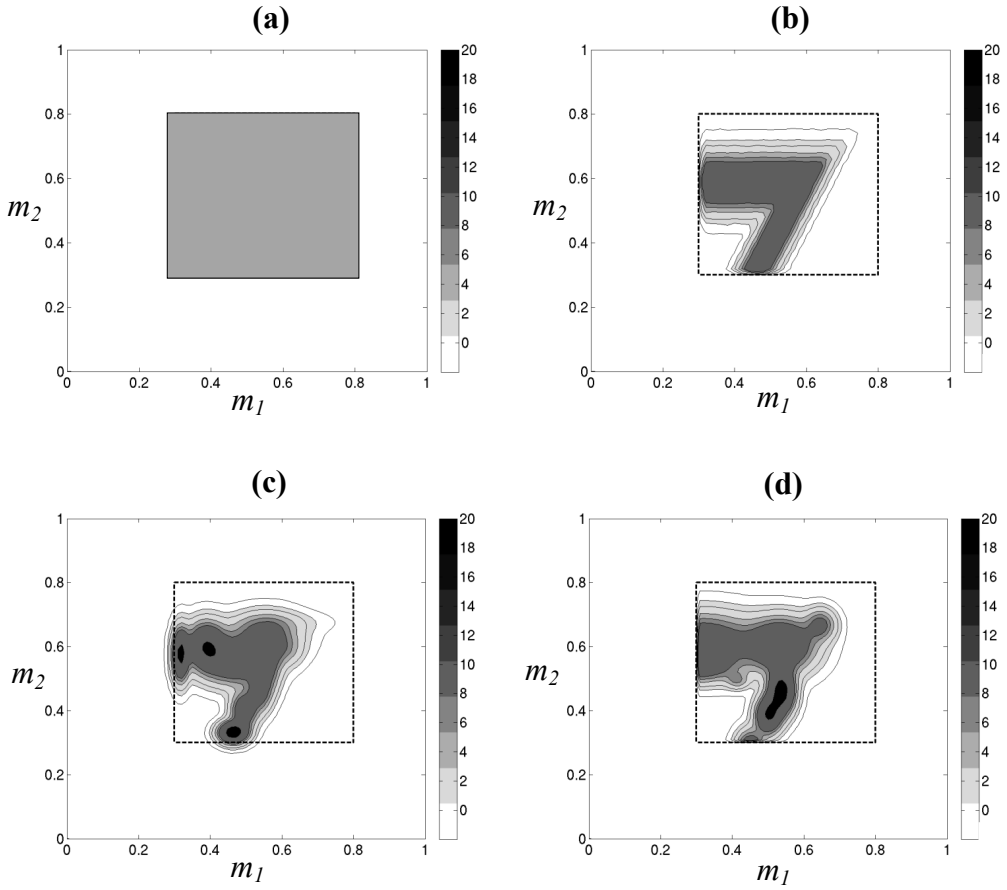

Figure 3.2: (a) The Uniform new prior PDF. (b) The posterior PDF obtained by MCMC sampling in the case of the Uniform new prior. This can be viewed as the 'true' posterior PDF for comparison. (c) The new posterior PDF obtained from the output of a neural network (MDN) trained with samples generated directly from the new prior, i.e., prior-specific training. (d) The new posterior PDF obtained by removing the old prior from the old posterior in Figure 3.1, and applying the new prior by prior replacement. In (b)-(d) the non-zero extent of the new prior is plotted with a stippled line. Prior-specific training has resulted in density appearing *outside* of these bounds.

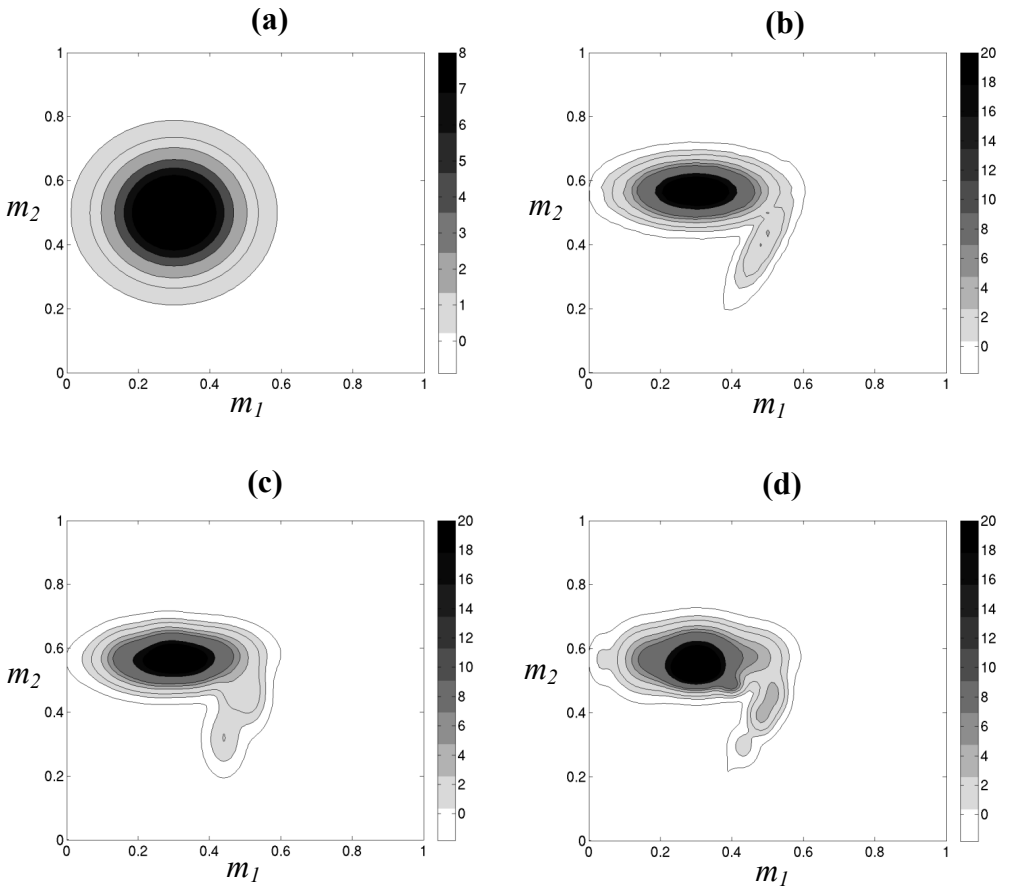

Figure 3.3: (a) The Gaussian new prior PDF. (b) The posterior PDF obtained by MCMC sampling in the case of the Gaussian new prior. This can be viewed as the 'true' posterior PDF for comparison. (c) The new posterior PDF estimate obtained from the output of a neural network (MDN) trained with samples generated directly from the new prior, i.e., prior-specific training. (d) The new posterior PDF estimate obtained by removing the old prior from the old posterior in Figure 3.1, and applying the new prior by prior replacement.

but with one such problem defined in each cell of a reservoir grid model. Then q would be equal to the number of cells M in the grid, which is typically over $\sim 10^5$ even for 2-D grids and can approach $\sim 10^9$ for 3-D grids (Buland and Omre, 2003; Shahraeeni et al., 2012). To demonstrate the usefulness of the this conclusion in this case we carried out an inversion test on a 2-D synthetic reservoir model using prior replacement. We created a 2-D model grid, with $X = 50$ and $Z = 50$, populated with the clay content by volume (m_1) and sandstone matrix porosity (m_2) parameters. Synthetic elastic parameter data $\mathbf{e}_i \; \forall \; i$, were created using the forward model $p(\mathbf{e}_i|\mathbf{m}_i)$ (see Appendix E). It was assumed that wells were present within the reservoir, down which m_1 and m_2 were known exactly. This well data was used to generate the (varying) prior information across the reservoir model in a realistic way (i.e., as commonly performed in industrial geophysics): Gaussian prior distributions were determined at each cell by kriging (a form of interpolation, see e.g., Olea (1999, pp. 7-17)) the known model parameters at the wells to each unknown cell using an appropriate covariance function and mean. The kriging estimate and variance were used as the Gaussian prior's mean and variance, respectively, in each cell. Inversion was carried out initially at each cell using a MDN trained with the broad old prior in equation 3.11, then the Gaussian priors were applied using prior replacement at each cell individually. Figure 3.4 depicts the model, the kriging-derived priors, and the inversion results. The inversion took ~ 200 seconds using the prior replacement method. Given that the grid contains $M = 50 \times 50 = 2500$ cells (and $q = M$) an equivalent result using prior-specific training would take $\sim 10^5$ seconds. Thus, even in this simple test, the prior replacement method provided a solution with a factor 10^3 gain in computational efficiency over comparable previous methods.

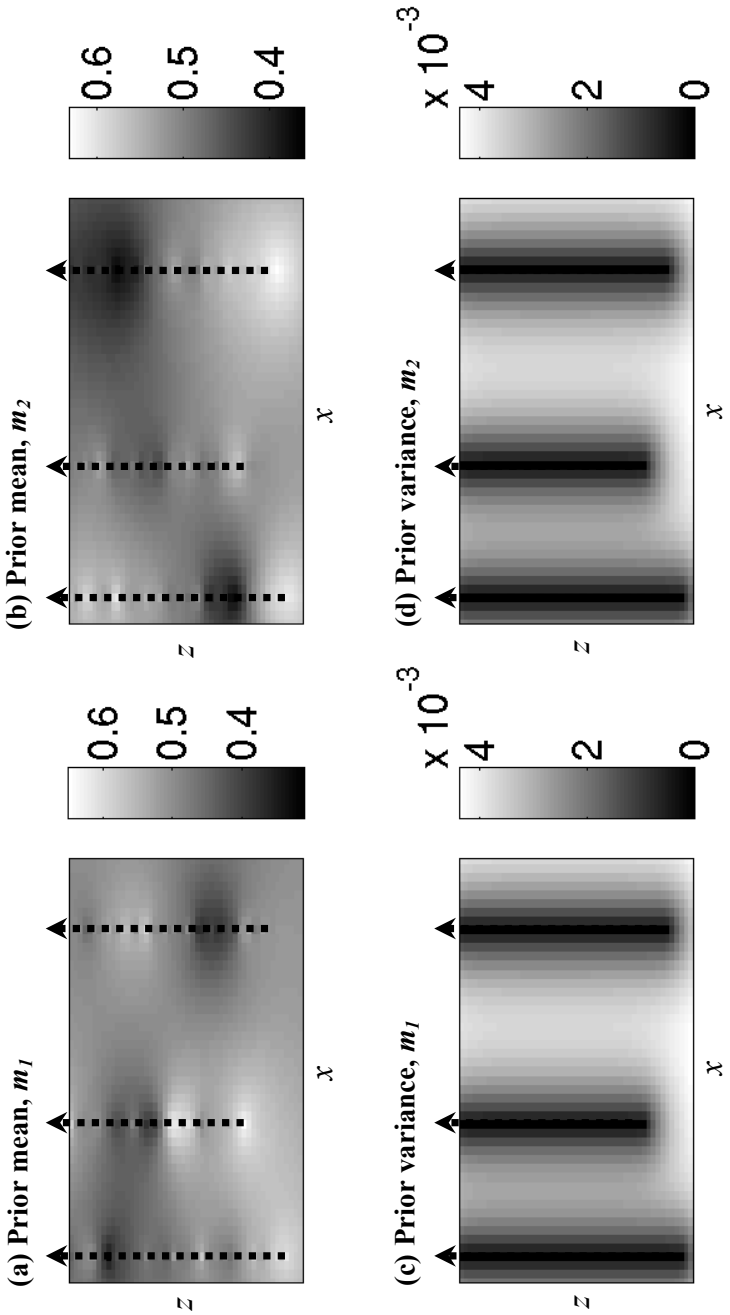

Figure 3.4: Synthetic inversion of elastic parameter data for the rock-physical (i.e., continuous geological) parameters clay content by volume (m_1) and sandstone matrix porosity (m_2), on a 2-D grid model. Gaussian prior PDFs over m_1 and m_2 were determined at each cell by kriging the known values of those parameters from well trajectories (marked with stippled lines): the kriging mean and variance were used as the prior Gaussian mean and variance at each unknown cell. Then prior replacement, using these prior distributions, was applied to the old posterior in Figure 3.1, to produce individual mixture density network (MDN) inversion results at each model cell. (a)-(b) and (c)-(d) show the prior mean and variance in each cell, respectively, obtained by kriging the well data for m_1 and m_2.

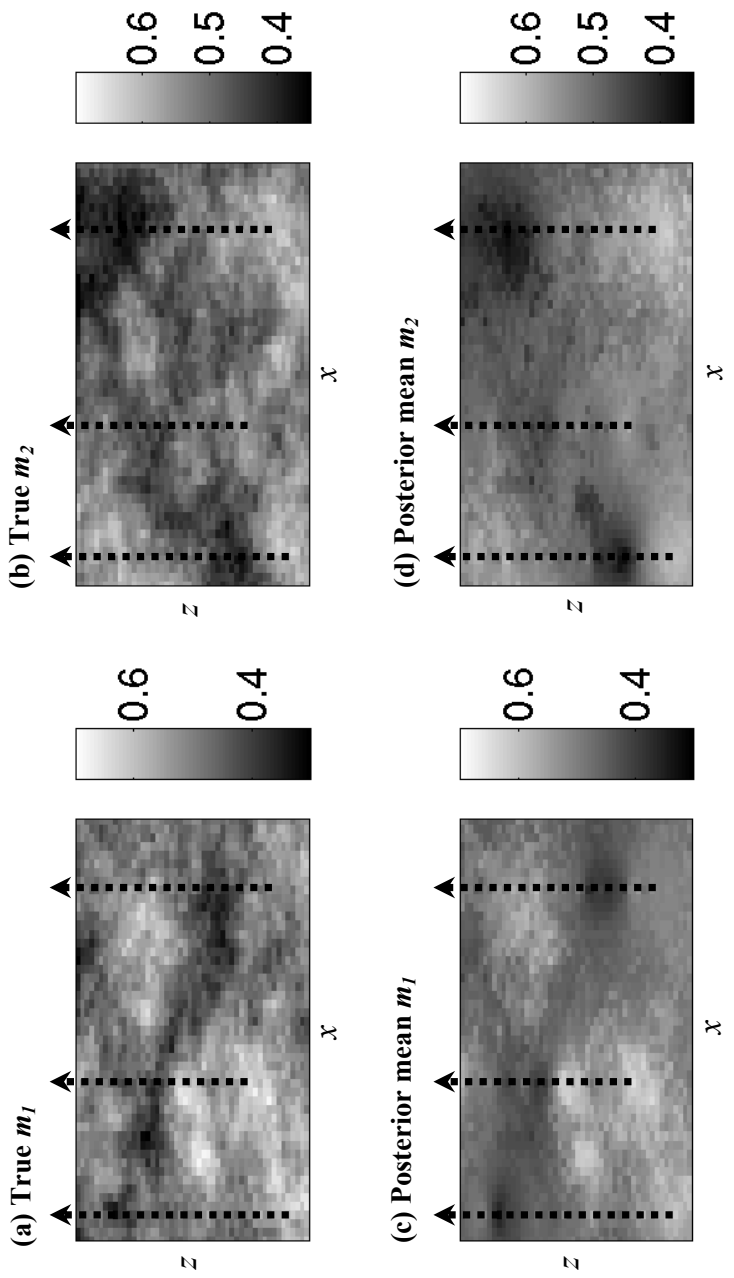

Figure 3.5: (a)-(b) The true rock physics parameters used to generate the synthetic elastic parameter data at each cell using equation E.4. (c)-(d) The posterior mean for m_1 and m_2 determined using prior replacement in MDN inversion (note these maps are smoother than the true model since we show the mean model estimator). The entire inversion method took ~ 200 seconds using prior replacement. An equivalent result using prior-specific training would take $\sim 10^5$ seconds.

3.8 Discussion

3.8.1 Numerical efficiency

We have shown that prior replacement can be useful for efficiently obtaining MDN inversion results with varying prior information. However, there is a significant computation required in the prior replacement method which is absent in prior-specific training. This is the normalisation step (equation D.15 or D.23), which must take place during every inversion for which the prior changes. While in the case of the results above it does not seem to slow the inversion greatly, as the number of dimensions of the model space grows, non-analytic integration will become significantly more costly. Using more advanced semi-analytical integration techniques for Gaussians (Drezner, 1992) may reduce this cost to some extent (we used only numerical integration here). We might also consider using only Gaussian priors for both training an MDN, and for use in the prior replacement methodology. As shown in Appendix D.5 this allows the normalising constant to be calculated analytically. However, this puts constraints on the form of the priors that may be non-physical. For example, assuming non-truncated Gaussians means assuming that the model space has non-zero probability everywhere; this might not be appropriate if we have hard constraints on model parameter values (e.g., in Shahraeeni and Curtis (2011) porosity must lie between 0 and 1).

Nevertheless, if we are able to perform efficient analytical normalisation (whether using the results derived in Appendix D.5 or some alternative parametrisation of posterior and priors) then prior replacement may be used for general Bayesian inverse problems (i.e., not MDN inversion) of much higher parameter space dimension. This could be very useful for problems where no closed form solution exists for the inverse. For example in subsurface reservoir studies, flow data measured at wells is often used to infer the permeability structure of the subsurface. Due to the sparsity of data in time and space the problem is ill-posed. Furthermore, the forward physics which is used to assess the likelihood of any particular model must be solved numerically at great computational cost using flow simulation. Thus, if MCMC methods are used to obtain an estimate of the posterior distribution over the subsurface permeability structure then it will be extremely computationally expensive. Due to the subjective nature of subsurface geological interpretation, however, prior information may change dramatically throughout the operational lifetime of a subsurface reservoir. In this

scenario the ability to change the prior distribution, a utility which prior replacement provides, may lead to hugely increased efficiency. This would be possible, given the discussion above, since Gaussian mixture models of the posterior distribution are often used in practice for such problems (Gu and Oliver, 2005).

It should also be noted that normalisation is not mandatory. If we do not require the absolute value of the probability, for example if we only wish to find the maximum-a-posteriori estimator or wish simply to sample from the GMM, then the normalisation step is not required and the new method becomes faster still. Furthermore, normalisation is unlikely to be an issue in problems which employ neural network inversion since the parameter space dimensionality is limited (typically to less than 10) by the amount of training data which may be processed in network training (Vapnik et al., 1994).

3.8.2 Quality of the posterior estimate

For MDN inversion, prior replacement always returns a distribution which is consistent with the final (i.e., the new) prior that is applied. This is not necessarily the case for prior-specific training because it fits the posterior distribution using Gaussians of finite size, and hence for example will always position some density outside of the bounds of a Uniform prior. This failure is clear in the results of prior-specific training in Figure 3.3(c) where non-zero contours of the posterior lie in the zero probability regions of the new prior. By contrast, Figure 3.3(d) shows that when prior replacement is used, no density is emplaced outside of the bounds of the new prior since the multiplication of prior and likelihood is explicit. Thus we envisage that prior replacement could be used in future for MDN inversion to ensure that the 'hard' bounds of a prior are enforced in the final posterior estimate.

Figure 3.3(c) shows a poor quality result using prior-specific training. Here the diagonally orientated lobe of low probability observed in the true posterior in Figure 3.3(b) is poorly resolved in Figure 3.3(c). The prior replacement result in Figure 3.3(d) resolves this feature better. This phenomena may be attributed to the data used to train the MDN in each case. Specifically, in prior replacement samples are spread more equally across the parameter space due to the broader old prior that is used. As such, the variance of the posterior distribution may be better reproduced. By contrast in prior-specific training, sampling was concentrated around a peak in the posterior induced by the new, more informative, prior. Thus we might expect

that the regions of high probability and hence the mean of the posterior would be better reproduced in this case. Indeed, it does appear that the high probability lobe in Figure 3.3(c) compares more favourably in shape to that in Figure 3.3(b) than does the lobe in Figure 3.3(d). Thus, it appears that some aspects of the posterior estimates may be improved by prior replacement (compared to prior-specific training), whereas other aspects appear to be more poorly estimated. Thus, again we envisage that prior replacement could be used in the future to enhance the results of MDN inversion, where prior-specific training gives inadequate results. For example, it may be desirable that the posterior is better resolved within a certain region of the model space, thus we might use prior replacement to ensure that the training data contains more samples from this important region by using an appropriate old prior.

A more sophisticated analysis of the quality of the results is clearly necessary if the effect of prior replacement on the posterior estimate is to be understood in greater depth. To this end we have performed an empirical analysis of the effect of prior replacement on an inverse problem where the posterior is modelled by a single Gaussian kernel. This analysis is presented in Appendix F. The results support our hypothesis that the effect of prior replacement on the quality of the posterior estimate is due to the distribution of samples used to estimate the old posterior (i.e., the form of the old prior). They also show that the effect is comparable, but not identical, to that of the Monte-Carlo technique of importance sampling (see e.g., Bishop, 2006, pp. 532-536), which suggests that at least an intuitive understanding of the effects of prior replacement may be borrowed from that method. The results in Appendix F also suggest that prior replacement could be used to manipulate the quality of the posterior estimate for general Bayesian inverse problems. For example, one may wish to better constrain the variance of the posterior in a Bayesian inverse problem solved using MCMC. Then, similarly to those results obtained in MDN inversion in Figure 3.3, this could be achieved by initially assuming a broad old prior and then, using prior replacement, emplacing the appropriate PDF as the new prior. However, more work is required to formalise such an operation.

There are a number of additional sources of error in the methodology which we have not yet described explicitly. The first of these arises from the fact that the neural network which is used to emulate the mapping between data and parameter space has a number of parameters which must be defined manually. The most important of these is the number of weights in the network, which controls the complexity of

the mapping. As explained in section 2.7.3 imposing too much complexity may lead to over-fitting, whilst the opposite may lead to bias (a poor fit to training data). Also, the GMM itself is an imperfect model of the posterior since it has a finite number of kernels. Furthermore, training is performed using optimisation which may be subject to local convergence. Thus careful effort must be made to validate the neural network model before combining it with prior replacement. In general, one should be aware that it is much more difficult to predict the accuracy of the resulting posterior probabilities obtained using network inversion (especially coupled with prior replacement) than those obtained using MCMC (which is guaranteed to converge to the correct new posterior after a sufficient number of samples is made).

3.9 Summary

We have derived expressions which allow the analytical computation of Bayesian posterior probability distributions with a variety of prior distributions using the method of prior replacement, particularly for Gaussian mixture models (GMMs). This procedure involves inverting for an 'old' posterior, determined by a likelihood PDF and old prior PDF, and then analytically replacing the old prior with a 'new' prior. We have shown that prior replacement can be a useful method for varying the prior distribution within the result of mixture density neural network (MDN) inversion. This avoids the computationally expensive step of MDN re-training at every instance that prior information changes (i.e., the MDN only has to be trained once). Prior replacement will then return a correct posterior provided the new prior distribution is non-zero only within the non-zero region of the old prior. We have also shown that prior replacement can be used as a tool to improve the results of MDN inversion in terms of certain statistical characteristics of the posterior distribution.

We have shown that we can use neural network inversion to obtain the mapping $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i)\ \forall\ i$ in a subsurface model grid, and then use prior replacement to vary the prior $p(\mathbf{m}_i)$ distribution implicit within the resulting posterior estimates with respect to i. The operation can be easily generalised to discrete geological parameters. Thus this achieves one of the objectives set out in section 1.7: neural network inversion can be used to do geological inverse problems where $p(g_i)$ (or $p(\mathbf{m}_i)$) varies with i. However, as argued in section 1.7 this can only be considered a valid solution to the geological inverse problem where $p(\mathbf{g}) = \prod_{i=1}^{M} p(g_i)$, which implies that there is

no (prior) correlation between the geological parameters in different cells. However, in the next chapter we will show that prior replacement permits the results of neural network inversion to be integrated into stochastic geological inversion schemes where $p(\mathbf{g})$ is defined using full conditionals, and as such admits such correlation.

It should be noted that other methods, similar to neural network inversion, exist for the determination of $\mathbf{e}_i \rightarrow p(\mathbf{m}_i|\mathbf{e}_i) \ \forall \ i$. For example, Grana and Della Rossa (2010) used a Gaussian mixture model to model the *whole* joint density $p(\mathbf{m}_i, \mathbf{e}_i)$ (and thus the required conditional $p(\mathbf{m}_i|\mathbf{e}_i)$ could be calculated from this for any given $\mathbf{e}_i$) to solve a similar problem to that of Shahraeeni and Curtis (2011). There is no reason why prior replacement cannot be used to modify the outputs of such methods, with similar efficiency savings. For example, prior replacement could be used to modify the prior implicit in the results of the method of Grana and Della Rossa (2010), without having to re-estimate the joint density $p(\mathbf{m}_i, \mathbf{e}_i)$.

References

Bailer-Jones, C., and K. Smith (2010), *Combining probabilities, Tech. Rep.GAIA-C8-TN-MPIA-CBJ-053*, Max Planck Institute for Astronomy, Heidelberg.

Bishop, C. M. (1994), *Mixture density networks, Tech. Rep.NCRG/94/0041*, Dept. of Computer Science and Applied Mathematics, Aston University.

Bishop, C. M. (1995), *Neural networks for pattern recognition*, Oxford University Press.

Bishop, C. M. (2006), *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc.

Buland, A., and H. Omre (2003), Bayesian wavelet estimation from seismic and well data, *Geophysics*, *68*(6), 2000–2009.

Drezner, Z. (1992), Computation of the multivariate normal integral, *ACM Transactions on Mathematical Software (TOMS)*, *18*(4), 470–480.

Grana, D., and E. Della Rossa (2010), Probabilistic petrophysical-properties estimation integrating statistical rock physics with seismic inversion, *Geophysics*, *75*(3), O21–O37.

Gu, Y., and D. Oliver (2005), History matching of the PUNQ-S3 reservoir model using the ensemble Kalman filter, *SPE journal*, *10*(2), 217–224.

Marion, D. P. (1990), Acoustical, mechanical, and transport properties of sediments and granular materials, Ph.D. thesis, Stanford University, Department of Geophysics.

McLachlan, G., and D. Peel (2004), *Finite mixture models*, Wiley.

Michie, D., D. J. Spiegelhalter, and C. C. Taylor (1994), *Machine learning, neural and statistical classification*, Ellis Horwood.

Olea, R. (1999), *Geostatistics for engineers and earth scientists*, Kluwer Academic Boston.

Petersen, K. B., and M. S. Pedersen (2006), *The matrix cookbook*, Technical University of Denmark.

Shahraeeni, M. S. (2011), Inversion of seismic attributes for petrophysical parameters and rock facies, Ph.D. thesis, The University of Edinburgh.

Shahraeeni, M. S., and A. Curtis (2011), Fast probabilistic nonlinear petrophysical inversion, *Geophysics*, *76*(2), E45–E58.

Shahraeeni, M. S., A. Curtis, and G. Chao (2012), Fast probabilistic petrophysical mapping of reservoirs from 3D seismic data, *Geophysics*, *77*(3), O1–O19.

Tarantola, A. (2002), *Inverse problem theory: Methods for data fitting and model parameter estimation*, Elsevier Science.

Vapnik, V., E. Levin, and Y. Le Cun (1994), Measuring the VC-dimension of a learning machine, *Neural Computation*, *6*(5), 851–876.

Yin, H., A. Nur, and G. Mavko (1993), Critical porosity: A physical boundary in poroelasticity, in *International journal of rock mechanics and mining sciences & geomechanics abstracts,* pp.805–808, Pergamon.

Chapter 4

Exact sampling for geological inversion

4.1 Overview

In section 1.6 we described how the geological inversion problem can be solved using stochastic Monte-Carlo methods. Stochastic methods estimate $p(\mathbf{g}|\hat{\mathbf{e}})$ (equation 1.17) by obtaining a set of samples from it, and then use those samples to characterise it (Mosegaard and Sambridge, 2002). Characterisation might include probability estimation, or calculating point estimates or moments of the distribution. Obtaining samples from a distribution, for which one only knows the unnormalized density or probability, may be achieved using Markov-chain Monte-Carlo (MCMC) methods. However, MCMC methods can suffer from bias issues since they rely on the assumption that the distribution of a chain of correlated samples (which the methods produce) converges to the posterior distribution within a *finite* set of samples; generally there are no proofs that suggest this is true. In this chapter, we derive a recursive algorithm for computing a decomposition of the posterior into a set of conditional distributions, which permits direct sequential sampling of $\mathbf{g}$ from $p(\mathbf{g}|\hat{\mathbf{e}})$. Thus this allows independent, rather than correlated, samples to be made from the posterior, and no assumptions need to be made regarding convergence. Henceforth this is referred to as *exact* sampling, and the method may be a useful alternative to MCMC sampling methods.

4.2 Introduction

The derivation of the recursive algorithm assumes the local likelihood property, and that $p(\mathbf{g})$ is defined using the full conditional distribution (equation 1.13). In practice, it is often assumed that dependency within equation 1.13 can be limited to a certain subset of the surrounding cells called the neighbourhood of cell i, $Ne(i)$. In this case the full conditional can be written as

$$p(g_i|\mathbf{g}_{\mathcal{H}\setminus i}) = p(g_i|\mathbf{g}_{Ne(i)}) = \frac{p(g_i, \mathbf{g}_{Ne(i)})}{p(\mathbf{g}_{Ne(i)})}. \tag{4.1}$$

It is important to note that the definition of the neighbourhood as such means that a cell is not a member of its own neighbourhood, $i \notin Ne(i)$. As with equation 1.13 a single, duplicate full conditional for all M cells in the grid is then used to define the prior $p(\mathbf{g})$ as a whole, which is to say that $p(g_i|\mathbf{g}_{Ne(i)})$ is invariant to i (except at the edge of the grid, where simple modifications can be made to compensate for any absent neighbours specified by $Ne(i)$). Henceforth we refer to this property, i.e., that we can specify the prior using a full conditional as in equation 4.1, as the *local prior property*. The derivation of the recursive algorithm is also dependent upon the assumption of this property.

The method developed here is quite general: it may be applied to any problem which fulfils the local prior and likelihood properties, and not just the geological inverse problem. However, it cannot be used as a useful alternative to MCMC methods for problems which do not fulfil these properties. For example, it could not be used to solve the elastic inverse problem where the local likelihood property is certainly not fulfilled (see equation 1.8).

The ability to specify the prior using a full conditional is central to the derivation of the algorithm, but the limitation of the conditional dependency to a certain range of cells is not (theoretically $Ne(i)$ may be any size). However, we will show later that the computational cost of the algorithm scales exponentially with the size of $Ne(i)$ and the (minimum) dimension of the model grid (i.e., X, Y or Z for a 3-D grid). Thus in practice limitations on the size of $Ne(i)$ must be considered; such assumptions about limited (conditional) spatial dependency in $\mathbf{g}$ are often made in geological inversion (and other spatial inverse problems), so this does not obviate practical application of the algorithm. However, the effect of the dimension of the model grid on computational cost is not so easily reduced, and we therefore also develop

an approximate version of the recursive algorithm to insure that the algorithm is computationally feasible for large grids. We also find that the cost of the algorithm scales with $|\mathcal{G}|$, thus there must also be limitations to the size of the sample space of the geological parameters but, again, these are not so strong as to prevent the practical use of the algorithm.

Before describing the methodology, in section 4.3 we briefly describe the notation used in rest of this chapter. In section 4.4 we discuss the specification of $p(\mathbf{g})$ using full conditionals in detail. Then in section 4.5 we further describe the convergence problems of MCMC methods, since this motivates the construction of the exact sampling method here. In section 4.6 we first describe the decomposition of the geological posterior $p(\mathbf{g}|\hat{\mathbf{e}})$, which can be used to sample from the geological posterior exactly. We then derive the recursive algorithm, which calculates the terms in this decomposition, for a 2-D grid (section 4.6.1). After a discussion of the algorithm's computational cost, we discuss possible limitations on $Ne(i)$ and $|\mathcal{G}|$, and define the approximate algorithm which permits application to realistically-sized grids (section 4.6.3). Finally we apply the approximate algorithm to a 2-D synthetic geological inversion problem in section 4.7, and compare the results to that of Gibbs sampling, a MCMC algorithm.

4.3 Notation

The notation used in this chapter follows that used in the introduction, except the elastic parameter vector used within the synthetic data demonstration of the recursive algorithm does not include density, i.e., $\hat{\mathbf{e}}_i = [I_P, I_S]_i$. We will demonstrate the method for sampling of discrete geological parameters $\mathbf{g} \in \mathcal{G}^M$, only. However, we make use of continuous geological parameters $\mathbf{m}$, to construct a forward relationship between g_i and $\hat{\mathbf{e}}_i$ for the synthetic demonstration of the method (section 4.7). Note that, as per the workflow illustrated in Figure 1.1, it is assumed that $\hat{\mathbf{e}}_i$ represents the results of deterministic elastic inversion in this chapter. However, as in the previous chapter, in order to simplify notation we discontinue the use of the hat symbol, and use $\mathbf{e}_i$ in place of $\hat{\mathbf{e}}_i$ in this chapter. A summary of the notation used in this chapter is given in Appendix H.3.

4.4 Full conditionals and Markov random fields

A set of M non-restricted full conditionals (equation 1.13) does not necessarily correspond to a valid geological prior distribution $p(\mathbf{g})$, and the same is true for a set of M full conditionals restricted by the local prior property in equation 4.1 (Besag, 1974).

A joint distribution $p(\mathbf{g})$ only gives rise to a valid set of M full conditionals if the so-called *positivity* condition is fulfilled. This requires that, if the individual marginal probability of each g_i is non-zero over its entire sample space (i.e., $p(g_i) > 0 \; \forall \; g_i \; \in \; \mathcal{G}, \; \forall \; i$ which we assume to be the case here), then the joint probability of all the g_i variables must be non-zero over their entire joint sample space (i.e., $p(\mathbf{g}) > 0 \; \forall \; \mathbf{g} \; \in \; \mathcal{G}^M$). The positivity condition on the joint distribution requires that the full conditionals themselves obey $p(g_i|\mathbf{g}_{Ne(i)}) > 0 \; \forall \; g_i \; \in \; \mathcal{G}, \; \forall \; i$. The necessity of the positivity requirement can be motivated by attempting to apply Brook's lemma (Brook, 1964) to calculate $p(\mathbf{g})$ from full conditionals containing zero probabilities (see e.g., Rue and Held (2005, pp.30-31)).

Even if positivity is fulfilled, an arbitrary set of full conditionals does not necessarily define a *valid* joint probability distribution $p(\mathbf{g})$. This is because the full conditionals may not be self-consistent (one may again motivate this by using Brook's lemma to determine the joint probability with arbitrarily-chosen full conditional distributions). Hammersley and Clifford (1971) were the first to describe the necessary conditions on $p(\mathbf{g})$ which must be met for it to yield a set of full conditionals with a certain neighbourhood structure. The Hammersley-Clifford theorem as proven by Besag (1974) states that $p(\mathbf{g})$ must factorise over sets of indices called 'cliques'. A clique is defined as a set of indices, $\Lambda = [\lambda_1, ..., \lambda_{|\Lambda|}]$, where each element $\lambda_i \in \{Ne(\lambda_q), \; \forall \; \{q \in 1, ..., |\Lambda|\} \backslash i\}$: in words, it is a set comprising indices which are all neighbours of each other. $p(\mathbf{g})$ must factorise over all cliques defined by the chosen neighbourhood structure on the grid. This ensures that when full conditionals are calculated from the joint distribution (i.e., using equation 4.1), the correct neighbourhood dependency structure is induced. In turn, this implies that the prior must have the form

$$p(\mathbf{g}) = \prod_{j=1}^{C} f_j(\mathbf{g}_{\Lambda_j}) \tag{4.2}$$

where C is the number of cliques on the grid, f_j are functions of the cliques, and $\mathbf{g}_{\Lambda_j}$ is the set of all g_i variables within the j^{th} clique. This equation defines a Markov

random field (Besag, 1974) and embodies the *factorisation* condition which must be met by the joint distribution to yield full conditionals with a certain neighbourhood structure. Since the full conditionals are derived from the joint distribution it is possible to determine the appropriate factorisation conditions on the full conditionals which yield a valid joint distribution (Besag, 1974).

In the case of geological inversion, we stipulated that the full conditionals are invariant to i, that is that we specify the prior by a single, duplicate full conditional (except at the edges of the grid). Regardless, the full conditional(s) must still meet the above conditions. Appropriate full conditional probabilities which meet these conditions can be derived from training images (e.g., Varma and Zisserman (2003)). It is easy to see that the factorisation requirement is irrelevant if the neighbourhoods are not restricted as in equation 1.13 since then each g_i variable is a neighbour of all others (then the cliques are the size of the grid, and no factorisation is required). However, if the neighbourhoods are limited in extent (which, as suggested in section 4.2, we must apply for computational efficiency), this factorisation requirement reduces the flexibility of the full conditional distribution (Besag, 1974). Thus we employ a more pragmatic approach in section 4.7 to obtain the full conditionals used to demonstrate the recursive algorithm.

Typical neighbourhood structures are illustrated in Figure 4.2. A common choice for $Ne(i)$ is a square centred on i. These neighbourhoods can be defined by the length of the square's sides, S (see Figure 4.2(b)-(c)). Simple modifications are made to such neighbourhoods when i is close to boundaries (i.e., where there are no neighbours beyond boundaries). We will henceforth consider only such square neighbourhoods for derivation of the method.

4.5 Convergence problems of MCMC methods

In MCMC methods a chain of correlated samples is created from a target distribution. If the chain is long enough the set of samples converges in distribution to the target distribution (Gilks et al., 1996). For example, if we wish to sample from the geological posterior $p(\mathbf{g}|\mathbf{e})$ we could use the archetypal MCMC algorithm, the Metropolis-Hastings algorithm (Metropolis et al., 1953; Hastings, 1970) summarised in algorithm 1.

The 'proposal distribution' q used in algorithm 1 (equation 4.3) is chosen on

Algorithm 1 The Metropolis-Hastings algorithm for sampling from $p(\mathbf{g}|\mathbf{e})$, where $\mathcal{U}[\mathcal{L}]$ is a Uniform distribution which is non-zero only over the set $\mathcal{L}$.

Obtain the initial ($t = 0$) sample $\mathbf{g}^{t=0} \sim \mathcal{U}[\mathcal{G}^M]$;
For $t = 1, 2, ..., n$

Obtain a candidate by sampling $\mathbf{g}'$ from the 'proposal distribution':

$$\mathbf{g}' \sim q(\mathbf{g}'|\mathbf{g}^{(t-1)}); \tag{4.3}$$

Calculate probability α of transitioning to the candidate:

$$\alpha = \min\left\{1, \frac{p(\mathbf{g}'|\mathbf{e}) \cdot q(\mathbf{g}^{(t-1)}|\mathbf{g}')}{p(\mathbf{g}^{(t-1)}|\mathbf{e}) \cdot q(\mathbf{g}'|\mathbf{g}^{(t-1)})}\right\}; \tag{4.4}$$

With probability α set $\mathbf{g}^t = \mathbf{g}'$, otherwise set $\mathbf{g}^t = \mathbf{g}^{t-1}$;
End For

the basis of how well it promotes convergence to the desired distribution. Generally speaking, it should be as similar to the posterior distribution itself as possible (Haario et al., 1999). This is problematic since the posterior is not known a-priori, and using a proposal distribution which is very dissimilar to the target can lead to slow convergence. For example, consider a posterior PDF with one maximum, which has a small support within which most of the probability mass is contained. Because of its small support it might take many iterations of algorithm 1 to find the peak if we do not use a similar proposal distribution from which to draw candidates (this is the so-called Witch's Hat problem - see Kass et al. (1998)). This can be remedied by choosing a proposal distribution which promotes so-called random walk behaviour by making the proposal distribution conditionally dependent upon the current member of the chain $\mathbf{g}^{(t-1)}$ (as is explicitly written in equation 4.3); proposed candidates tend to be close to the current sample, and tend to be selected preferentially by equation 4.4 if they too have high probability. This heuristic enforces our intuition that high probability areas will be 'close' together within the parameter space, and encourages the chain to follow gradients toward regions of high probability.

The division in equation 4.4 implies that the normalization constant (of the geological posterior, equation 1.17) is never explicitly required for such an algorithm. The only requirement for convergence to the posterior distribution is that the Markov chain, which is induced by the use of the proposal distribution, be *irreducible*. Irreducibility means that all parts of the parameter space $\mathcal{G}^M$ may be reached by the

chain starting from any position in that space (Gilks et al., 1996). However, there is no assurance of convergence for finite n, and convergence is difficult to diagnose even if it occurs (Besag and Green, 1993). The chain may be biased towards its starting position so the initial part of the chain may exhibit 'transient' (non-stationary) behaviour. If the chain has converged it will exhibit some 'dynamic stationarity' and this in some cases may be used as a diagnostic of convergence. If the onset of stationarity can be detected, samples from this transient period (the so-called burn-in period) may be ignored in order to remove this bias from the ensemble.

Unfortunately, observing apparent dynamic stationarity over a finite set of samples does not imply that the ensemble has truly converged to the target distribution. This is problematic because it implies that the posterior distribution, which we estimate from the ensemble of samples, would be incomplete and biased (even if we remove the burn-in samples). For example, consider the case of a probability distribution having two distinct peaks, each with small support as in the example above. Suppose that the chain of samples were currently confined within one of those high probability peaks. The probability of moving to the other peak is low since not only must the proposal distribution produce a sample within the other peak, but the probability of transition to that sample may then also tend to be low (since the chain is already within a high probability region). This problem can be compounded by the use of local random walk proposal distributions if the probability of samples being chosen in between the peaks is low, since they may require that the chain traverse areas of low probability in order to move from one peak to another. This problem is similar to the problem of convergence to local maxima in optimisation problems (Saul and Roweis, 2003). However, in Bayesian inversion the objective is to determine the whole posterior distribution, and thus it is a problem if the chain becomes stuck in *any* maxima (whether it be global or local) since this implies that the rest of the distribution may be inadequately sampled. We cannot easily diagnose this problem because the chain may nevertheless exhibit dynamic stationarity within the region of the maxima. Thus in practice when we use MCMC techniques it is hard to guarantee convergence to the posterior and hence ensure that the ensemble of samples is unbiased (unless we have a good idea of what the posterior should be like a-priori).

There are many existing strategies which aim to detect or ensure convergence to the posterior by using heuristic rules to enhance mobility (or 'mixing') of the chain around the model space. Well-known examples include simulated annealing

(Kirkpatrick et al., 1983) and hybrid MCMC (Chen et al., 2001). Such methodologies have been used successfully in a wide range of applications but they do not ensure nor detect convergence: they only make it more probable that a non-biased estimate of the posterior will be found within a practical number of iterations.

To a large extent then, both making a choice of proposal distribution and our ability to correctly detect stationarity, depend on the form and strength of our prior information. As suggested in section 1.5.2, in geological inversion it is usual to specify much of the prior information in terms of relative spatial relationships between the variables in different grid cells, rather than in terms of values of the variable at absolute positions. In other words, probabilities are assigned to certain patterns or variations which occur across the model grid. The prior distribution $p(\mathbf{g})$ naturally has high variance: there are many possible configurations of $\mathbf{g}$ which contain relative relationships or patterns which are acceptable, but the euclidean distance between such configurations within $\mathcal{G}^M$ may be large. An example is if a variogram is used to describe porosity heterogeneity in a subsurface reservoir: generally there is a large range of configurations of porosity which would be consistent with any particular variogram (Olea, 1999, p.154). Furthermore, in section 1.5.2 we described how in general we must assume multi-modality in $p(\mathbf{g})$ (and also possibly $p(\mathbf{e}|\mathbf{g})$). Thus, by Bayes' rule (equation 1.17) we must expect multi-modality in $p(\mathbf{g}|\mathbf{e})$ (Shahraeeni et al., 2012). Thus the problems associated with bias in the convergence of MCMC sampling are highly relevant to the geological inverse problem (and, by extension, spatial inverse problems that invoke MCMC methods in general).

4.6 Methodology

In this chapter we derive a sampling methodology which avoids the use of MCMC sampling techniques altogether. The methodology estimates the conditional decomposition of the posterior distribution as

$$p(\mathbf{g}|\mathbf{e}) = \prod_{i=1}^{M} p(g_i|\mathbf{e}, \mathbf{g}_{<i}). \tag{4.5}$$

where $< i$ denotes the set of indices $1, ..., i-1$ which for $i = 1$ represents the empty set (such that $\mathbf{g}_{<i} = [g_1, g_2..., g_{i-1}]$). We refer to the $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$ distributions as the *partial conditionals*. Obtaining these distributions allows sequential sampling

from the geological posterior (Journel et al., 1998). This refers to the process of first sampling g_1 from $p(g_1|\mathbf{e})$, then g_2 from $p(g_2|\mathbf{e}, g_1)$, then g_3 from $p(g_3|\mathbf{e}, g_1, g_2)$ and so forth until g_M is sampled from $p(g_M|\mathbf{e}, \mathbf{g}_{<M})$, each time using the previously sampled $\mathbf{g}_{<i}$ variables as the conditioning variables. If each of the partial conditionals are of closed-form, then each can be sampled from exactly and the vector of samples for all cells $\mathbf{g}$ is itself an exact sample from the posterior. One then need only repeat the sequential sampling process to obtain another independent sample from the posterior; in this way we avoid the problems of convergence associated with the use of correlated MCMC sampling.

We use a recursive algorithm to determine the partial conditional distributions in closed-form based on the algorithm of Bartolucci and Besag (2002). Such recursive algorithms have their roots in hidden Markov chains (Baum et al., 1970; Scott, 2002) and have been applied to spatial inverse problems (Ulvmoen and Hammer, 2010). However, such methods require significant computational resources and as such in the past have only been applied to small problems (Friel et al., 2009). We believe that computational advances now make practical applications of these algorithms possible, when appropriate approximations are made to the conditional decomposition in equation 4.5. Indeed, Arnesen (2010) and Tjelmeland and Austad (2012) have already shown this to be true. However, the derivation of their recursive algorithm, and the required approximations for its practical application, are based on the representation of the posterior as a Gibbs potential (Friel and Rue, 2007). We present a more pragmatic approach and develop our approximation using a probabilistic terminology (developed initially by Bartolucci and Besag (2002)). Importantly this permits the exact sampling algorithm to be implemented easily, and adapted for use in geological inversion.

In the following sections, we develop the recursive algorithm for a 2-D grid specified, as usual, with Z rows and X columns and indexing as shown in Figure 4.1 (note the 'rows' and 'columns' terms are used to describe the z and x directions, respectively, for clarity in the derivation of the algorithm). The algorithm can easily be generalised to 3-D grids, or collapsed to 1-D grids. As stated above the recursive algorithm requires the assumption of the local geological likelihood (equation 1.2) and prior property (equation 4.1). We first derive the recursive algorithm and the topology of the partial conditionals which it calculates, in section 4.6.1 and 4.6.2, respectively. We then discuss its computational cost with respect to the parameters of the inversion, and the approximations which permit it to be applied to large grids,

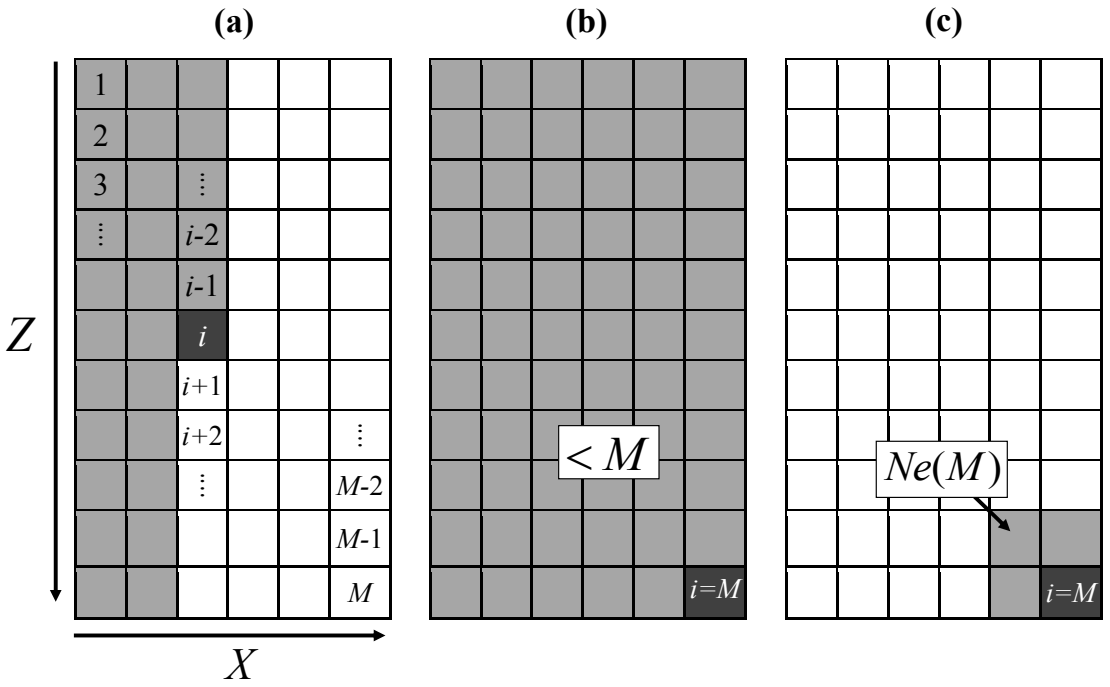

Figure 4.1: (a) Indexing of the 2-D grid with dimensions Z (number of rows) and X (number of columns). The total number of cells $M = Z \times X$. Also depicted is the dependency structure of the partial conditionals, $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$, in equation 4.5: the dark gray cell is the variate g_i, and the light gray cells are those containing the conditioning variables $\mathbf{g}_{<i}$. These distributions are also conditioned upon data in all cells, $\mathbf{e}$. (b) The dependency of $p(g_M|\mathbf{e}, \mathbf{g}_{<M})$ (i.e., when $i = M$). (c) When $i = M$ the set $\{< M\}$ must contain the neighbourhood of M, thus the dependency of $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$ is limited to the neighbourhood of M (one possible example of such a neighbourhood is shown here; other examples are shown in Figure 4.2).

in section 4.6.3.

4.6.1 The recursive algorithm

In order to determine the posterior and to sample from it efficiently, we develop a recursive algorithm based on the work of Bartolucci and Besag (2002). Set notation is used in the derivation, and brackets ($\{\}$) are used to enclose sets for clarity. As in the rest of the thesis, sets will be used to reference subsets of cells in the grid and their associated variables as a vector, for example $\mathbf{g}_{\{<4\}\setminus 1} = [g_2, g_3]$.

Our goal is to calculate the posterior distribution $p(\mathbf{g}|\mathbf{e})$ on the left hand side of equation 4.5 by evaluating the partial conditionals $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$ on the right hand side. These distributions can be found efficiently by using the recursive algorithm of Bartolucci and Besag (2002). Overall in the algorithm the partial conditionals are

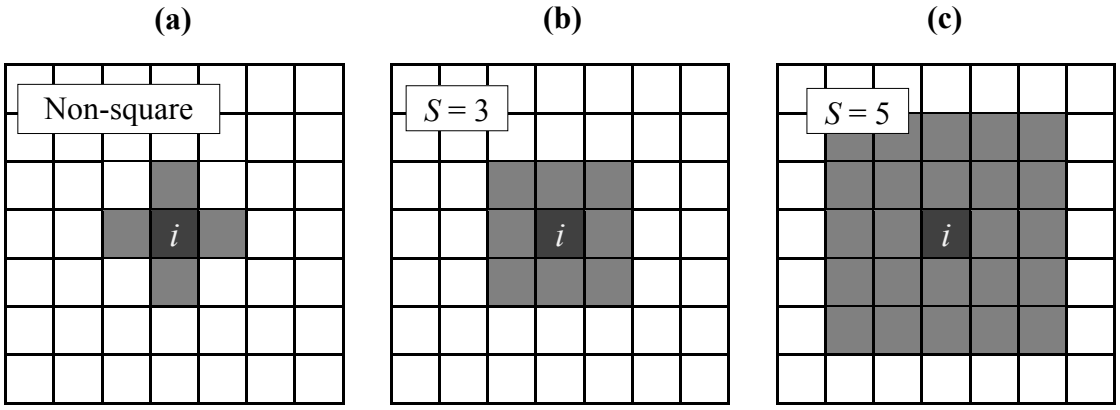

Figure 4.2: Possible neighbourhood arrangements. (a) A 'non-square' neighbourhood commonly used in image processing. (b) A square neighbourhood with side of length 3 cells (which we denote $S = 3$). (c) A square neighbourhood with $S = 5$.

calculated in the order $i = M, M-1, ..., 2, 1$. To calculate the partial conditional for cell i one must first calculate

$$p(g_i|\mathbf{e}, \mathbf{g}_{\{\leqslant k\}\backslash i}) \tag{4.6}$$

where

$$k = \max(Ne(i)). \tag{4.7}$$

Given the definition of k in equation 4.7, the set $\{\leqslant k\}\backslash i$ will contain the neighbourhood of i. Thus, because of the local prior property (equation 4.1), there can be no dependence on g_i variables outside of the neighbourhood in equation 4.6. Also there is no dependency on data apart from that located at cell i (in equation 4.6), because of the local likelihood property (equation 1.2). Thus we may rewrite equation 4.6 as

$$p(g_i|\mathbf{e}, \mathbf{g}_{\{\leqslant k\}\backslash i}) = p(g_i|\mathbf{e}_i, \mathbf{g}_{Ne(i)}), \tag{4.8}$$

and this expression can be decomposed, using Bayes' rule, into two terms:

$$p(g_i|\mathbf{e}, \mathbf{g}_{\{\leqslant k\}\backslash i}) = \mathcal{Z}_i p(\mathbf{e}_i|g_i) p(g_i|\mathbf{g}_{Ne(i)}) \tag{4.9}$$

where $p(\mathbf{e}_i|g_i)$ is the cell-wise geological likelihood, $p(g_i|\mathbf{g}_{Ne(i)})$ is the full conditional, and $\mathcal{Z}_i = \left(\sum_{g_i \in \mathcal{G}} p(\mathbf{e}_i|g_i) p(g_i|\mathbf{g}_{Ne(i)})\right)^{-1}$ is a normalising constant. If we assume that $p(\mathbf{e}_i|g_i)$ has been determined as a function of g_i and that we have obtained $p(g_i|\mathbf{g}_{Ne(i)})$, then equation 4.9 can be determined immediately. $\mathcal{Z}_i$ must be calculated by summation but this will be an undemanding task if both $\mathcal{G}$ and $|Ne(i)|$ are not

prohibitively large. Once equation 4.9 is determined then the identity

$$p(g_i|\mathbf{g}_{\{\leqslant j-1\}\setminus i},\mathbf{e}) = \left\{\sum_{g_j\in\mathcal{G}} \frac{p(g_j|\mathbf{g}_{<j},\mathbf{e})}{p(g_i|\mathbf{g}_{\{\leqslant j\}\setminus i},\mathbf{e})}\right\}^{-1} \tag{4.10}$$

from Bartolucci and Besag (2002), may be applied recursively, for $j = k, k-1, ..., i+2, i+1$. At $j = i+1$ the result gives the desired partial conditional at cell i. The application of this identity represents a secondary backward recursion within the algorithm. It should be understood that, since $i = M, M-1, ..., 2, 1$, the $p(g_j|\mathbf{g}_{<j},\mathbf{e})$ distributions in equation 4.10 will have been determined in the previous iterations. Consequently, the algorithm must be initiated at $i = M$ where the partial conditional term can be calculated immediately since the neighbourhood of cell M, $Ne(M)$ is entirely contained within the conditioning cells in the partial conditional (see Figure 4.1(c)), thus

$$p(g_M|\mathbf{g}_{<M},\mathbf{e}) = p(g_M|\mathbf{e}_M,\mathbf{g}_{Ne(M)}) = \mathcal{Z}_M p(\mathbf{e}_M|g_M)p(g_M|\mathbf{g}_{Ne(M)}), \tag{4.11}$$

where $\mathcal{Z}_M$ again denotes the normalizing constant required by Bayes' rule. Once $p(g_M|\mathbf{e},\mathbf{g}_{<M})$ is determined, then $p(g_{(M-1)}|\mathbf{e},\mathbf{g}_{<(M-1)})$ can be calculated and so forth, until all terms (partial conditionals) in the posterior decomposition (equation 4.5) are determined. Sequential sampling from $p(\mathbf{g}|\mathbf{e})$ can then be performed using the determined partial conditionals. The complete recursive algorithm is summarised in algorithm 2.

It should be noted that the conditional distributions as written in all equations above are strictly correct. However, there may be conditional independence from some of the written conditioning variables. We do not explicitly indicate this conditional independence here in order to make it clear that these distributions are conditioned by these variables (even if they may be conditionally independent); thus these distributions then cannot be confused for marginals over those conditioning variables. This is important because the domain of the numerator and denominator must be compatible for the division in equation 4.10 to be valid. A discussion of the conditional independence structure of the distributions is given in section 4.6.2, below.

Algorithm 2 can be used almost without modification for 3-D grids; only a change must be made to the indexing of the grid such that it runs over the third dimension

(in addition to the rows and columns of the 2-D case). A cubic 3-D neighbourhood structure would also have to be defined (using this indexing).

Algorithm 2 Recursive algorithm for a 2-D grid with Z rows and X columns with $M = Z \times X$ cells, and neighbourhood structure $Ne(i)$.

Calculate $p(g_M|\mathbf{g}_{<M}, \mathbf{e}) = \mathcal{Z}_M p(d_M|g_M)p(g_M|\mathbf{g}_{Ne(M)})$;
For $i = M-1, M-2, ..., 2, 1$
  Calculate $k = \max(Ne(i))$;
  Calculate $p(g_i|\mathbf{e}, \mathbf{g}_{\{\leqslant k\}\setminus i}) = \mathcal{Z}_i p(e_i|g_i)p(g_i|\mathbf{g}_{Ne(i)})$
  For $j = k, k-1, ..., i+2, i+1$
    Calculate the recursive identity

$$p(g_i|\mathbf{g}_{\{\leqslant j-1\}\setminus i}, \mathbf{e}) = \left\{ \sum_{g_j \in \mathcal{G}} \frac{p(g_j|\mathbf{g}_{<j}, \mathbf{e})}{p(g_i|\mathbf{g}_{\{\leqslant j\}\setminus i}, \mathbf{e})} \right\}^{-1};$$

  End For
  Retain $p(g_i|\mathbf{g}_{<i}, \mathbf{e})$;
End For

4.6.2 Details of conditional independence

The local prior and likelihood properties induce conditional independence in the partial conditional distributions. In terms of dependence upon the data, counter-intuitively, the partial conditionals must incorporate non-local likelihood information even if the local likelihood property is assumed. Consider the general case of the partial conditional at cell i, $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$; it is easy to show that because of the local likelihood property described by equation 1.2 we may rewrite this as being independent of the data $\mathbf{e}_{<i}$ which coincides with the conditioning $\mathbf{g}_{<i}$ variables. In mathematical terms $p(g_i|\mathbf{e}, \mathbf{g}_{<i}) = p(g_i|\mathbf{e}_{\geq i}, \mathbf{g}_{<i})$. By equation 1.2 it is also obvious that g_i is dependent upon e_i in the partial conditional. However, it is less obvious that g_i must also be dependent upon the data $\mathbf{e}_{>i}$, i.e., $p(g_i|\mathbf{e}, \mathbf{g}_{<i}) \neq p(g_i|e_i, \mathbf{g}_{<i})$. The reason for this is that $\mathbf{e}_{>i}$ yields information about $\mathbf{g}_{>i}$. Furthermore, the prior specifies correlation between the elements of $\mathbf{g}$. Thus $\mathbf{e}_{>i}$ must yield indirect information about g_i and therefore cannot be ignored in the partial conditional. Therefore, the recursive algorithm is designed to efficiently incorporate the non-local likelihood information (i.e., from $\mathbf{e}_{>i}$) into the calculation of the partial conditional distributions.

For the g_i variables, if we have assumed the local prior property, i.e., we assumed equation 4.1 with some $Ne(i)$, then the dependency in the partial conditional may be limited to a subset of $\mathbf{g}_{<i}$. This subset is determined by the global Markov property (Rue and Held, 2005, p. 24), which can be stated by supposing that we have three mutually exclusive subsets $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{S}$ of cells (indices) in the grid, and some neighbourhood structure for the g_i variables in the grid. The property then states that if starting at any cell in $\mathcal{A}$, and only by passing between neighbours, one cannot reach any cell in $\mathcal{B}$ without passing through a cell within $\mathcal{S}$, then $\mathbf{g}_{\mathcal{B}}$ is conditionally independent of $\mathbf{g}_{\mathcal{A}}$ given $\mathbf{g}_{\mathcal{S}}$. For square neighbourhood structures on a 2-D grid (e.g., Figure 4.1(a)), this can be used to show that the partial conditionals are limited in dependency such that it is possible to write

$$p(g_i|\mathbf{e}, \mathbf{g}_{<i}) = p(g_i|\mathbf{e}, \mathbf{g}_{J(i)}) \tag{4.12}$$

where

$$J(i) = \{j|j < i \wedge \max(Ne(j)) \geqslant i\}. \tag{4.13}$$

The resulting reduced dependency structure is demonstrated for a partial conditional in Figure 4.3 for the case of square neighbourhoods with $S = 3$ and $S = 5$. Application of the global Markov property to the distributions generated by equations 4.9 and 4.10 in the recursive algorithm yields distributions with similarly reduced dependency structure. Thus the domain of these distributions can be calculated, which permits the number of operations required to calculate equations 4.9 and 4.10 in algorithm 2 to be determined. This, in turn, permits the computational cost of the recursive algorithm to be estimated.

4.6.3 Computational limitations and approximations

Bartolucci and Besag (2002) derived an expression for the number of floating point operations required to calculate the partial conditionals, and hence determine the posterior, using algorithm 2 for the non-square neighbourhood structure illustrated in Figure 4.2(a). It can be derived by applying the conditional dependency structure discussed in section 4.6.2. We use a slightly modified version of the expression which gives an upper limit to the number of floating point operations required to calculate all the partial conditionals, for a 2-D grid with square neighbourhood structure of

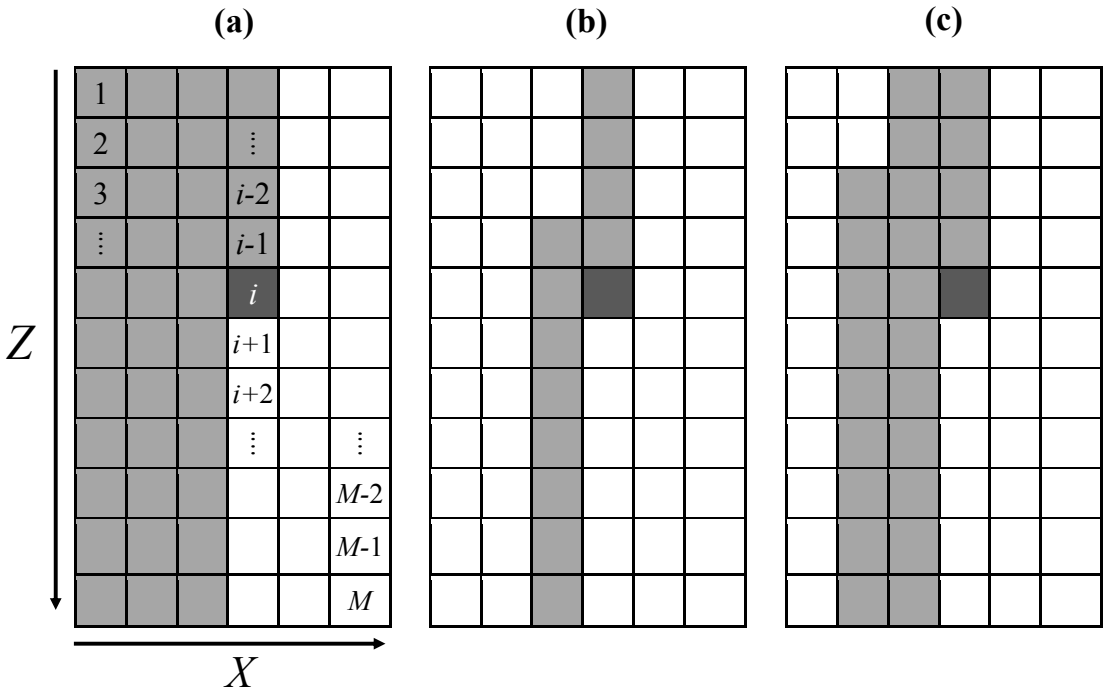

Figure 4.3: Illustration of conditional dependency structures of partial conditionals induced on a 2-D grid with square neighbourhood structures by the global Markov property. (a) The dependency structure of the partial conditional distribution, $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$, without consideration of conditional independence induced by a neighbourhood structure. (b) The dependency structure with consideration of the square neighbourhood structure (with side $S = 3$), i.e., $p(g_i|\mathbf{e}, \mathbf{g}_{<i}) = p(g_i|\mathbf{e}, \mathbf{g}_J)$ where $J = \{j|j < i \wedge \max(Ne(j)) \geqslant i\}$. (c) As for (b) but with a square neighbourhood with side $S = 5$.

side S, as

$$Z \times X \times S \times Z \times |\mathcal{G}|^{S \times Z} \tag{4.14}$$

where Z is the vertical dimension (i.e., number of rows), X is the lateral grid dimension (i.e., number of columns), S is the dimension of the square template, and $|\mathcal{G}|$ is the size of the sample space of g_i. Since the direction of indexing is arbitrary, Z and X are interchangeable (i.e., we could run the algorithm on a grid with indexing in the perpendicular direction to that in Figure 4.1). Thus if the dimensions are unequal then the direction should be chosen such that the lowest dimension appears in the exponent. Despite the exponentiation of $|\mathcal{G}|$ in equation 4.14, the size of the sample space would not cause computational problems for the recursive algorithm in many real applications. For example in geological inversion for reservoir parameters we often invert for discrete parameters such as lithology-fluid class. The number of such classes can be low (see, e.g., Rimstad and Omre (2010) where $|\mathcal{G}| = 4$) or geological considerations can allow us to reduce the number of classes by implementing 'nesting' of lithologies within one another.

Equation 4.14 illustrates the importance of the local prior property for efficient computation of the recursive algorithm: it is clear that since S appears in the exponent, the size of the square neighbourhood must be limited to permit efficient application of the algorithm. In many real applications S is assumed to be quite low (see, e.g., Rimstad and Omre (2010) where $S = 3$). Thus this limitation does not obviate the practical application of the algorithm. Unfortunately, however, realistically sized grids have a minimum dimension of at least hundreds of cells (Caers, 2005). Since this number appears in the exponent (Z in equation 4.14), it is clear that the algorithm, as presented, would be computationally infeasible even with sufficiently low S and $|\mathcal{G}|$ parameters. This motivates us to define an approximation that permits the algorithm to be applied to realistically sized grids by reducing the number which appears in the exponent. To do this we henceforth assume that we have indexing as defined in Figure 4.1(a). Then, roughly speaking, the approximation is to take smaller bands of the grid and run algorithm 2 on these bands.

More precisely, for each row in the grid $z = 1, 2, ..., Z-1, Z$, the set of rows $l(z) = \{\max(z-a, 1), ..., z, ..., \min(z+a, Z)\}$ are selected, where a is the so-called approximation parameter. Note that by definition $l(z)$ ignores non-existent rows. This defines a so-called sub-grid for each z, denoted $[\mathbf{g}^{*z}, \mathbf{e}^{*z}]$, whose elements are defined by $[g_i, e_i] \in [\mathbf{g}^{*z}, \mathbf{e}^{*z}] \; \forall \, i : \mathcal{R}(i) \in l(z)$, where the operator $\mathcal{R}(i)$ returns the row

(i.e., z) to which the cell with index i belongs. Algorithm 2 is run on each $[\mathbf{g}^{*z}, \mathbf{e}^{*z}]$ sub-grid. Once run, each cell in each sub-grid has a partial conditional distribution associated with it. For each sub-grid, only the partial conditionals for the cells in row z are retained as approximations to the partial conditionals in the *complete* grid. In mathematical terms we set $p(g_i|\mathbf{g}_{<i}, \mathbf{e}) \approx p(g_i|\mathbf{g}^{*z}_{<i}, \mathbf{e}^{*z}) : z = \mathcal{R}(i)$. These are approximate because they are only dependent upon cells within the range of the smaller sub-grid used in algorithm 3, and likewise only conditioned upon data in that grid. Figure 4.4(a) illustrates the use of sub-grids for calculating the approximate partial conditionals, and Figure 4.4(b) illustrates the resulting dependency structure in one of these distributions. In effect, the approximation reduces the range at which data, e_i, is incorporated into the calculation of the partial conditionals in one direction (e.g., here the range is limited in the vertical z direction). Also the range of the conditioning cells (in terms of the g_i variable) is limited. The result of this approximation is that the computational upper bound in equation 4.14 is reduced to

$$Z \times X \times S \times a \times |\mathcal{G}|^{S \times a}. \tag{4.15}$$

This approximate algorithm is summarised in algorithm 3. An analogue of this algorithm for 3-D grids would consist of defining cubic sub-grids (rather than rectangular sub-grids, as in the 2-D case) and then running the 3-D version of algorithm 2 on these sub-grids. However, expansion to three dimensions may be computationally expensive since the exponent in equation 4.15 would become $S \times a \times b$ where the approximation parameters a and b now describe the (limited) size of the cubic sub-grid in two dimensions.

Algorithm 3 Approximate recursive algorithm for a 2-D grid with Z rows and X columns with $M = Z \times X$ cells, and approximation parameter a. The operator $\mathcal{R}(i)$ returns the row to which the cell with index i belongs.

For $z = 1, 2, ..., Z-1, Z$
  Select rows $l(z) = \{\max(z-a, 1), ..., z, ..., \min(z+a, Z)\}$;
  Define subgrid $[g_i, e_i] \in [\mathbf{g}^{*z}, \mathbf{e}^{*z}]\ \forall\ i : \mathcal{R}(i) \in l(z)$;
  Run algorithm 2 with sub-grid $[\mathbf{g}^{*z}, \mathbf{e}^{*z}]$ to obtain $p(g_i|\mathbf{g}^{*z}_{<i}, \mathbf{e}^{*z})\ \forall\ i : \mathcal{R}(i) \in l(z)$
End For
Retain approximations $p(g_i|\mathbf{g}_{<i}, \mathbf{e}) \approx p(g_i|\mathbf{g}^{*z}_{<i}, \mathbf{e}^{*z}) : z = \mathcal{R}(i)$;

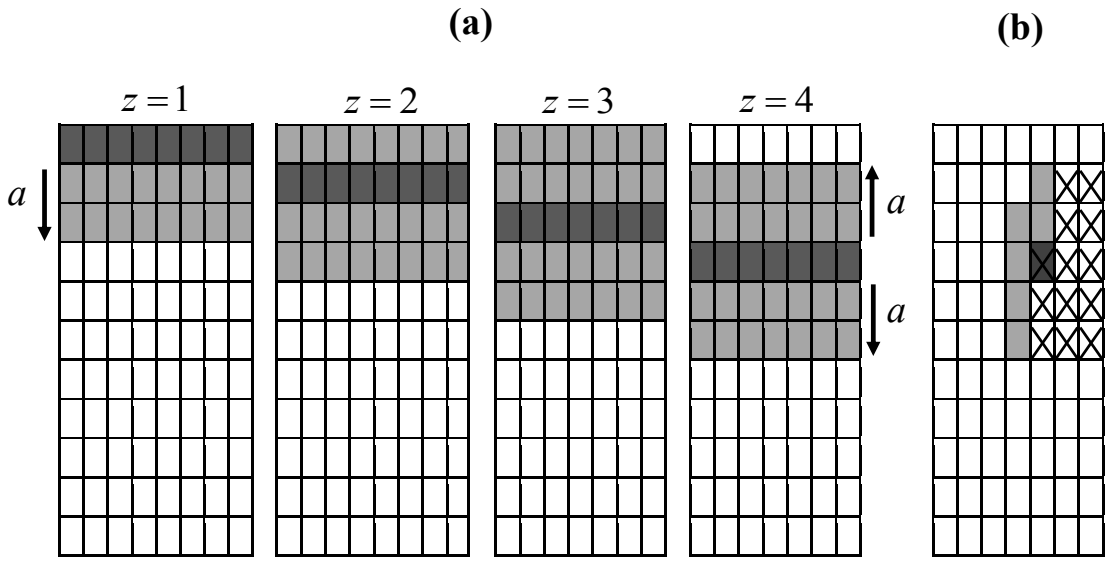

Figure 4.4: Illustration of the approximation (algorithm 3) to the recursive algorithm (algorithm 2) with approximation parameter $a = 2$. The full recursive algorithm is run on sub-grids centred on each row of the complete grid. Sub-grids comprise a rows above and below the current row of the complete grid. When a rows do not exist in either direction the sub-grid is truncated to include only the available rows. Partial conditionals are determined for each cell of each sub-grid. (a) Shows the sub-grids centred on rows $z = \{1, 2, 3, 4\}$, where gray cells are members of the sub-grids. The partial conditionals determined in the dark gray cells (i.e., for row z of each sub-grid) are retained as approximations to $p(g_i|\mathbf{g}_{<i}, \mathbf{e})$, and are thus used for exact sampling from $p(\mathbf{g}|\mathbf{e})$. (b) The dependency structure of the resulting approximate partial conditionals. The dark gray cell is the cell containing the variate, g_i. The light gray cells are those containing the conditioning g_i variables (note we have taken into account the conditional independence implied by the global Markov property given in equations 4.12-4.13). The cells containing crosses are those containing data which are involved in the evaluation of the corresponding partial conditional.

4.7 Synthetic application

We tested the approximate recursive algorithm by applying it to a synthetic geological inverse problem involving the inversion of elastic parameter estimates for lithology-fluid class. A categorical geological parameter g_i is used to represent lithology-fluid class, where

$$g_i \in \mathcal{G} = \{\text{Shale, Gas-sand, Brine-sand}\}. \tag{4.16}$$

Two 2-D grids were populated with this univariate geological parameter using a simple geological process model. The model generated channel shapes and overbank deposits. These were filled with brine-sand and emplaced within a shale lithology. Gas was then emplaced in some of the channels, in a manner consistent with gas-saturation in such geological features (i.e., obeying gravitational ordering). One of these grids, shown in Figure 4.5, was used to determine the full conditional distributions and hence the prior $p(\mathbf{g})$ (thus this grid is henceforth referred to as the training image). The other, shown in Figure 4.6(a), was used to generate synthetic elastic parameter $\mathbf{e}$ data (thus it is henceforth referred to as the target grid). These data will be inverted using the proposed approximate recursive algorithm. The elastic parameter data was generated by considering each cell in the grid independently and using a probabilistic forward model, $p(\mathbf{e}_i|g_i)$ to generate collocated S- and P- wave impedances $\mathbf{e}_i = [I_P, I_S]_i \; \forall \; i$.

To define $p(\mathbf{e}_i|g_i)$, we began by choosing an appropriate rock-physics forward function, the Yin-Marion shaley-sand model, which can predict the P- and S-wave impedances ($\mathbf{e}_i$) for a given shale-sand mixture and pore fluid. Here three rock-physical parameters were allowed to vary: clay volume content m_1, sandstone matrix porosity m_2, and water saturation m_3, such that $\mathbf{m}_i = [m_1, m_2, m_3]_i$. A component of random Gaussian noise was added to the output of the rock-physics forward-function, thus it could be written as a PDF, $p(\mathbf{e}_i|\mathbf{m}_i)$. Full definitions of the Yin-Marion shaley-sand model and $p(\mathbf{e}_i|\mathbf{m}_i)$ are given in Appendix E.

This forward relationship only permits the generation of $\mathbf{e}_i$ once the rock-physical parameters $\mathbf{m}_i$ are specified. Thus the next part of defining $p(\mathbf{e}_i|g_i)$ required definition of a relationship between g_i and $\mathbf{m}_i$. This should be uncertain (probabilistic) as we would expect a lithology-fluid class to have a range of possible different rock-physical parameters (Avseth et al., 2005). Thus, each lithology-fluid class (in

Table 4.1: Table describing bounds used to define $p(\mathbf{m}_i|g_i)$.

Lithology-fluid class	Clay content by volume (m_1)	Sandstone matrix porosity (m_2)	Water saturation (m_3)
Shale	$[0.20, 0.40]$	$[0.20, 0.40]$	$[1.00, 1.00]$
Gas sand	$[0.00, 0.20]$	$[0.20, 0.40]$	$[0.05, 0.60]$
Brine sand	$[0.00, 0.20]$	$[0.20, 0.40]$	$[0.60, 1.00]$

equation 4.16) was assigned a distribution $p(\mathbf{m}_i|g_i)$ describing the probability of the rock physical parameters for that particular class. We described these relationships using simple bounds $[lower, upper]$ on the possible values of each rock physical parameter, for each lithology-fluid class (see table 4.1). The probability distribution of the rock-physical parameters within these bounds was Uniform.

With $p(\mathbf{m}_i|g_i)$ and $p(\mathbf{e}_i|g_i)$ defined (that is, mappings between g_i and $\mathbf{m}_i$, and between $\mathbf{m}_i$ and $\mathbf{e}_i$ are defined), the full probabilistic forward function can be defined as

$$p(\mathbf{e}_i|g_i) = \int_0^1 \int_0^1 \int_0^1 p(\mathbf{e}_i|\mathbf{m}_i)p(\mathbf{m}_i|g_i)d\mathbf{m}_i. \tag{4.17}$$

This distribution can be sampled from without performing the integration analytically (which would be very difficult given the form of the rock physics forward model described in Appendix E) by sampling sequentially first $\mathbf{m}_i$ from $p(\mathbf{m}_i|g_i)$ and then $\mathbf{e}_i$ from $p(\mathbf{e}_i|\mathbf{m}_i)$. Thus we may sample from the distribution and obtain the synthetic data $\mathbf{e}_i$ from the lithology-fluid class g_i in each cell in the target grid. The resulting data are shown in Figure 4.6(b) and (c); as can be observed, the distribution of sand facies in Figure 4.6(a) is just discernible in these plots, however, there is little or no visual distinction between gas- and brine- sand facies.

We have shown that it is possible to sample $\mathbf{e}_i$ from $p(\mathbf{e}_i|g_i)$ given g_i using equation 4.17. However, equation 4.9 in the recursive algorithm requires that we have access to $p(\mathbf{e}_i|g_i)$ as *a function of* g_i. To obtain this, for all i, we begin by defining the prior distribution $p(g_i)$ to be Uniform (over $\mathcal{G}$). Sampling g_i from this distribution and then sampling $\mathbf{e}_i$ from equation 4.17 allows us to sample from the joint distribution $p(\mathbf{e}_i, g_i) = p(\mathbf{e}_i|g_i)p(g_i)$. Such samples can be used to estimate $p(\mathbf{e}_i, g_i)$ parametrically, and this parametric distribution can be used to obtain the desired distribution as a function of g_i by fixing $\mathbf{e}_i$ and calculating $p(\mathbf{e}_i|g_i) = p(\mathbf{e}_i, g_i)/p(g_i)$. The results are shown in Figure 4.7 for all cells in the target grid. This parametric estimation is computational simple since g_i is discrete and is small in terms of its

sample space (i.e., $|\mathcal{G}| = 3$ from equation 4.16), and can be performed by fitting a Gaussian mixture model over the elastic parameter space ($\mathcal{E}$) for each lithology-fluid class.

It is important to note that we could not have obtained the likelihood without estimating $p(\mathbf{e}_i, g_i)$ first. Initially, although we could sample from $p(\mathbf{e}_i|g_i)$, we did not know it *parametrically*. The latent (or 'nuisance') parameters $\mathbf{m}_i$ prevented us from doing so, thus sampling was required. However, the estimation of the joint distribution (and hence the sampling) need only be done once, and obtaining the likelihood distribution at each cell in the target cross section only requires fixing $\mathbf{e}_i$ at the appropriate value in $p(\mathbf{e}_i|g_i) = p(\mathbf{e}_i, g_i)/p(g_i)$.

We must apply another prior (the full conditional) to $p(\mathbf{e}_i|g_i)$ within the recursive algorithm (equation 4.9). This represents a prior replacement calculation (as described in Chapter 3), which is the algebraic replacement of a prior implicit within one posterior distribution by a new, different prior distribution using Bayes' rule, to form a new posterior. To avoid undefined probabilities in the new posterior arising from division by zero in this calculation, the Uniform prior distribution $p(g_i)$ used to estimate $p(\mathbf{e}_i, g_i)$ must have non-zero probabilities wherever we expect the new, replacing prior to have non-zero probabilities (this is equivalent to the support condition as defined in section 3.4). In this case the Uniform distribution over (the entirety) of the discrete sample space $\mathcal{G}$ satisfies this requirement.

Equivalently, neural network inversion (as in Chapter 3) could have been used here to determine $\mathbf{e}_i \rightarrow p(g_i|\mathbf{e}_i)\ \forall\ i$, and the prior replaced by the full conditional in equation 4.9 (using prior replacement). However, since g_i is discrete in this case, mixture density network (MDN) inversion is not required and the simple method of parametric estimation of the joint distribution $p(\mathbf{e}_i, g_i)$ using Gaussian mixture models can be used instead.

In this demonstration we chose the neighbourhood for the prior full conditional to be square with $S = 3$; the actual distribution $p(g_i|\mathbf{g}_{Ne(i)})$ was derived from the training image, by visiting each cell in the training image grid which had appropriate neighbours available, and counting the occurrences of each conditional event. This method does not necessarily return a full conditional which is consistent with a valid joint distribution $p(\mathbf{g})$ joint over all i in the grid (see section 4.4). Nevertheless, positivity can be ensured by simply adding a small number to any zero probabilities calculated in the full conditional this 'event-counting' method. Factorisation can be ensured by using equation 4.2 to define $p(g_i|\mathbf{g}_{Ne(i)})$ (i.e., as a product of functions

defined with the appropriate cliques as their domains), and using the training image to constrain these functions, rather than the probabilities directly (Varma and Zisserman, 2003). However, by definition, the cliques are smaller than the neighbourhoods thus this method requires that such *valid* full conditionals must have a more parsimonious parametrisation than simply specifying every probability in $p(g_i|\mathbf{g}_{Ne(i)})$ independently (as the event-counting method does). Thus we found that attempting to use a full conditional which does definitely satisfy the factorisation condition cannot contain as much (prior) information as those returned by the event-counting method, and hence does not produce sufficiently realistic inversion results for the geological parameters.

Therefore we simply assume that the full conditional probabilities (with correction for positivity) obtained using the event-counting method are *approximately* correct. This leads to equation 4.10 being approximate, which in practice means that equation 4.10 yields probability distributions which are not normalised. Typically the error in probability mass is < 0.1 and we simply re-normalize equation 4.10 to correct for this. This approximation is an added source of error in the results of the recursive algorithm. However, below we compare its results to those obtained using an MCMC algorithm which uses exactly the same prior information, and show that it compares favourably.

4.7.1 Results

With the likelihood distributions (as a function of g_i) at each cell and the full conditional determined, the recursive algorithm can be applied and the approximate partial conditionals calculated. The approximation length used was $a = 4$. Once the partial conditionals have been found, independent samples from the geological posterior can be determined rapidly. Using the recursive algorithm to find the partial conditionals took approximately 10800 seconds on a standard desktop computer. Making each independent sample from the approximate posterior (specified using the partial conditionals) took approximately 0.1 second.

An ensemble of 1×10^4 samples from the posterior was made using the approximate recursive algorithm 2. Figure 4.8 shows four example realisations from the ensemble. The ensemble of realisations was used to calculate the posterior cell-wise marginal probability of gas-sand occurrence (i.e., $p(g_i = \text{gas-sand}|\mathbf{e})$ at each cell) as an example of the kind of statistics that are then calculable. This is plotted in

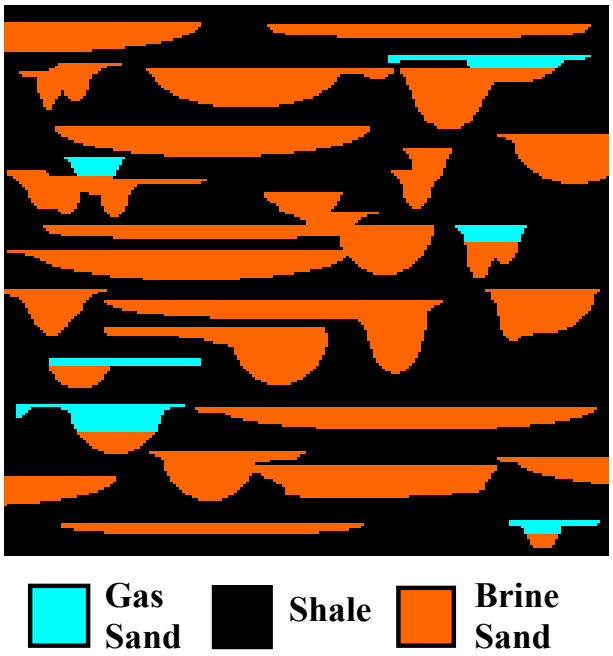

Figure 4.5: Training image grid used to obtain the probabilities in the full conditional, $p(g_i|\mathbf{g}_{Ne(i)})$. The training image represents a 2-D cross section from the 3-D result of a geological process model. It contains sand-filled channels with overbank deposits, emplaced within shale. Gas has been injected into some of the channels.

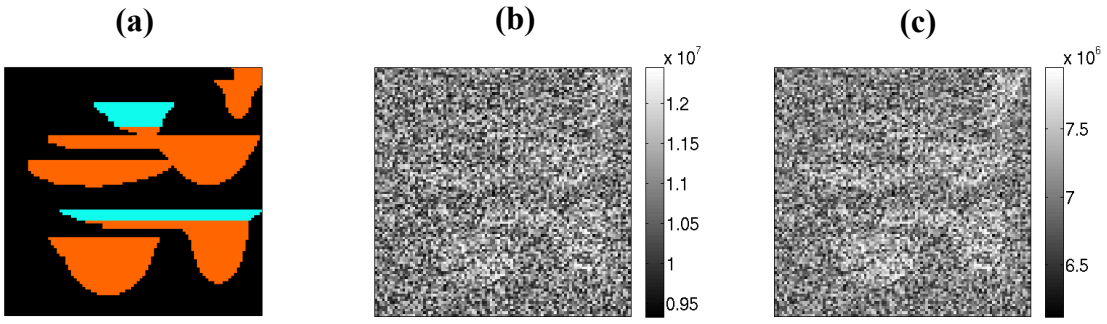

Figure 4.6: (a) The target grid. (b) and (c) show S- and P-wave impedance data ($\mathbf{e}$), respectively, generated using the probabilistic forward model (the Yin-Marion shaley-sand model).

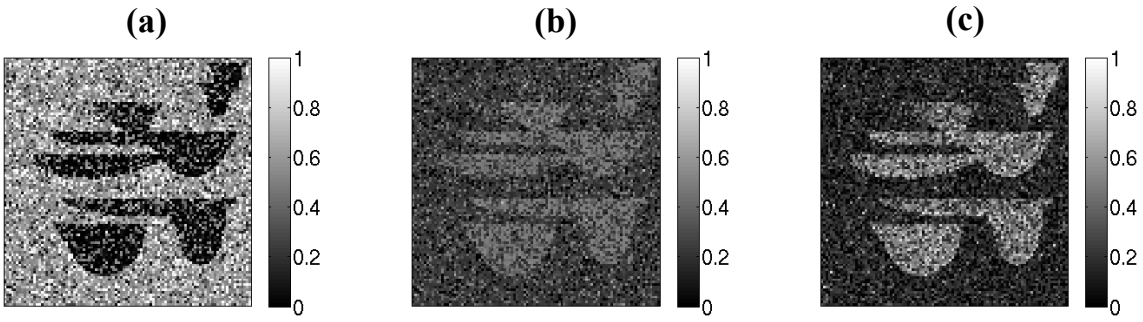

Figure 4.7: The likelihood of (a) shale, (b) gas-sand, and (c) brine-sand, determined using a parametrised version of $p(g_i, \mathbf{e}_i)$. The likelihoods are normalized such that (in each cell) we have $p(\mathbf{e}_i|\text{shale}) + p(\mathbf{e}_i|\text{gas-sand}) + p(\mathbf{e}_i|\text{brine-sand}) = 1$.

Figure 4.9 along with the target grid for comparison.

4.8 Comparison to Gibbs sampling

The results show that reasonable results can be obtained using the recursive algorithm. The realisations in Figure 4.8 from the approximate posterior exhibit similarities to the target section in Figure 4.9(a). The cell-wise posterior mean of gas-sand occurrence in Figure 4.9(c) is consistent both with the true gas-sand distribution in Figure 4.9(b), and the uncertainty which we might expect: it is nearly certain that the two gas accumulations exist, but there remains some uncertainty as to their exact extent. The quality of the estimate in Figure 4.9(c) compared to the information content of the likelihood in Figure 4.7 shows the additional value of the prior information contained in Figure 4.5 and embodied in the full conditionals which define $p(\mathbf{g})$.

These approximate results are somewhat difficult to appraise since we do not have an exact geological posterior result with which to compare. An alternative estimate for the posterior can be made using MCMC methods. However, such a method of sampling may suffer from the convergence and bias problems described in section 4.5 which motivated us to develop the algorithm in the first place. Nevertheless, we used an MCMC methodology called Gibbs sampling (Geman and Geman, 1993) to obtain samples from the posterior and hence obtain an alternative posterior estimate for comparison.

The Gibbs sampler, summarised in algorithm 4, uses the distribution $p(g_i|\mathbf{g}_{Ne(i)}, \mathbf{e})$

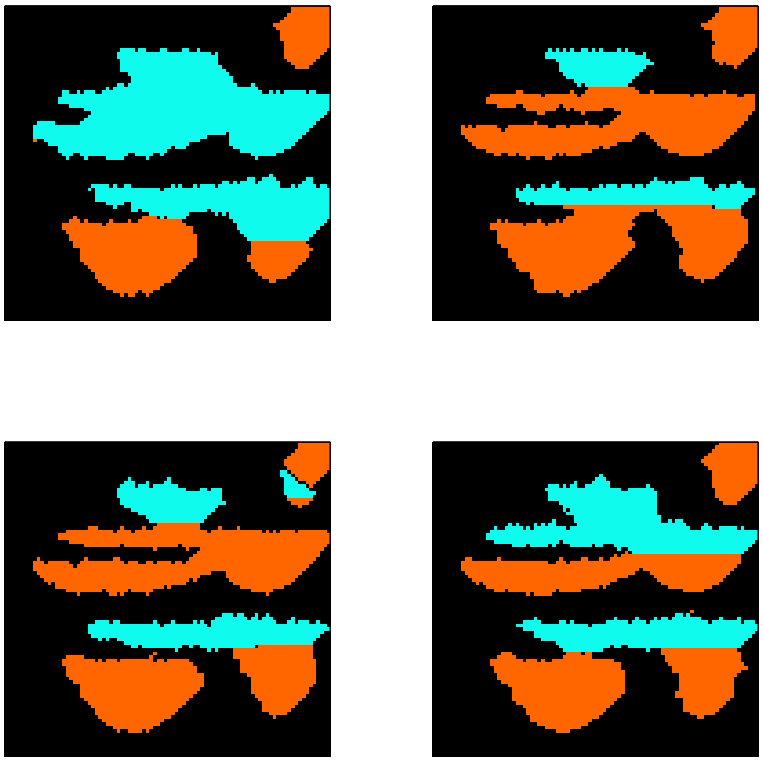

Figure 4.8: Four realisations (samples) of $\mathbf{g}$ from the geological posterior $p(\mathbf{g}|\mathbf{e})$, obtained using the approximate recursive algorithm.

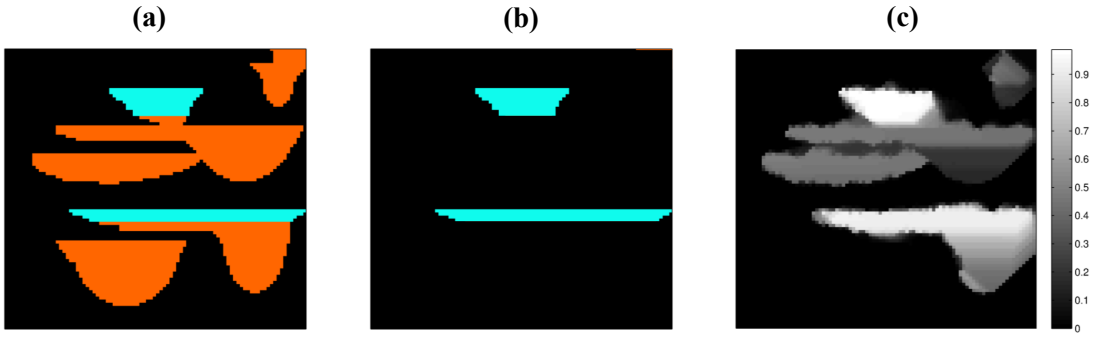

Figure 4.9: (a) The target grid used to generate the elastic parameter data ($\mathbf{e}$). (b) The true distribution of gas-sand in this grid for comparison. (c) The posterior cell-wise marginal probability of gas-sand occurrence (i.e., $p(g_i = \text{gas-sand}|\mathbf{e})$ at each cell) generated from the ensemble of samples from the posterior $p(\mathbf{g}|\mathbf{e})$, obtained using the approximate recursive algorithm.

to update g_i at each cell at a time in the grid. This distribution is similar to the full conditional but it is dependent on the data $\mathbf{e}$. It can be derived from the full conditional (equation 4.1) using Bayes' rule as

$$p(g_i|\mathbf{g}_{Ne(i)}, \mathbf{e}) = \frac{p(\mathbf{e}_i|g_i)p(g_i|\mathbf{g}_{Ne(i)})}{\int_{\mathcal{G}} p(\mathbf{e}_i|g_i)p(g_i|\mathbf{g}_{Ne(i)})dg_i}. \tag{4.18}$$

where $p(g_i|\mathbf{g}_{Ne(i)})$ is the full conditional and $p(\mathbf{e}|\mathbf{g})$ is the joint geological likelihood (defined in equation 1.3). This is a simple expression which may be calculated immediately if the cell-wise likelihood (as a function of g_i) and full conditional are known. It can be shown that algorithm 4 is a special case of the Metropolis-Hastings algorithm (and as such it will converge to the target distribution eventually if the chain is irreducible) where the proposal distribution q is $p(g_i|\mathbf{g}_{Ne(i)}, \mathbf{e})$ and the probability of transition is always unity (Geman and Geman, 1993). Importantly, it can be proved that, if the full conditionals satisfy the positivity condition, the chain is irreducible and hence eventual convergence is assured (Robert and Casella, 2004, p. 376). The Gibbs sampler in effect removes the need to choose q by using the prior as the proposal distribution, which is a common approach in MCMC methods (e.g., Tarantola, 2002) where the prior is available.

Because the Gibbs sampler is a random walk MCMC algorithm which only updates one cell at a time, it moves very slowly around the posterior and thus convergence to the target distribution can be slow (Belisle, 1998; Van Dyk and Park, 2008). Furthermore the risk of becoming stuck in a maxima is increased because changes in the current state are incremental (i.e., they are only ever at a single cell). This problem can, to a certain extent, be addressed by rerunning the algorithm from different starting points. However this may be of limited use if the parameter space is large (Brooks and Gelman, 1998), and this approach does not in any case solve the fundamental problem which is the difficulty in ensuring that the Gibbs sampler will be able to visit all important parts of the parameter space within a practical time period, and hence produce a chain of samples which will determine the posterior without bias.

We used the same cell-wise likelihood distributions and full conditional (used in the approximate recursive algorithm to obtain the results in section 4.7.1) in algorithm 4 to sample from $p(\mathbf{g}|\mathbf{e})$ using Gibbs sampling. Initially we ran the algorithm for 1×10^8 iterations which took approximately 9×10^9 seconds. We removed many

Algorithm 4 Gibbs sampling algorithm for sampling from $p(\mathbf{g}|\mathbf{e})$, where $\mathcal{U}[\mathcal{L}]$ is a Uniform distribution which is non-zero only over the set $\mathcal{L}$.

Obtain initial sample $\mathbf{g}^{t=0} \sim \mathcal{U}[\mathcal{G}^M]$;
For $t = 1, 2, ..., n$
  Set $\mathbf{g}^t = \mathbf{g}^{t-1}$;
  Choose a cell i at random in the grid, $i \sim \mathcal{U}[1, M]$;
  Sample from $g_i' \sim p(g_i'|\mathbf{g}^t_{Ne(i)}, \mathbf{e})$;
  Set $g_i^t = g_i'$;
  Retain $\mathbf{g}^t$;
End For

of the resulting realisations by only retaining a sample every 4×10^6 iterations (this process of 'thinning' removes highly correlated samples). From the 25 realisations retained, a cell-wise posterior probability of gas-sand occurrence was calculated. This estimate and the final realisation retained are plotted in Figure 4.10(a)-(b). As discussed above the Gibbs sampler has the tendency, in practice, to become 'stuck' in probability maxima (and therefore can yield biased results). Thus we re-initiated the algorithm with a different random starting point $\mathbf{g}^{t=0}$ and repeated the procedure. The results are plotted in Figure 4.10(d)-(e); note that slightly more realisations (30) were retained after thinning in the second run of Gibbs sampling. The two results are remarkably different. It seems that each has become stuck in a different probability maxima. This conclusion is reinforced when we inspect the sequential difference between the retained realisations (plotted in Figure 4.10(c) and (f)): the realisations change greatly at the start of the algorithm but as the number of iterations increases these changes become increasingly small. Indeed even when the first chain in Figure 4.10 was run for 10^9 iterations (taking approximately 24 hours) there was little change in the retained realisations (e.g., only one accumulation of gas was ever realised).

4.9 Discussion

It is clear from the results of the previous section that the Gibbs sampling result cannot be trusted - it is clearly highly biased toward the starting point because the chain induced is not practically recurrent. For example if we ran the algorithm just once and got the upper results in Figure 4.10 we would only detect one of the accumulations of gas, while the lower results in Figure 4.10 contradict this conclusion.

However, Gibbs sampling does deliver individual realisations which are more consistent in some ways with the true model: for example, the shape of the channels is better defined in the Gibbs sampling results (the recursive algorithm produces 'rough' channel edges). It should also be noted that if we take the two Gibbs results together they are consistent with the results of the approximate recursive algorithm (which shows that both gas accumulations are almost certainly present simultaneously). The results of the recursive algorithm are therefore consistent with those of Gibbs sampling, but they seem to be more reliable since there is no bias induced by the starting point of the algorithm to (local) probability maxima.

Errors in the results of the recursive algorithm may be attributed to the approximations used to determine the partial conditionals. Not only will this approximation error be a function of the approximation parameter (a) but also of the characteristics of the posterior distribution itself (controlled by the forward relation and the prior). We have not derived a method to obtain the approximation error a-priori, or even a bound on it. This is a general problem with such approximation methods (Friel and Rue, 2007). Even the rigorously-derived, graph-theory based approximation of Arnesen (2010), cannot predict or bound the error a-priori. Thus either (i) an extensive empirical study of the relationship between approximation quality and those parameters mentioned above should be made, or (ii) the approximation should be rephrased in order to admit some way of finding a bound on the error. It is not clear how (ii) could be accomplished, thus option (i) seems a more likely starting-point for future work.

Another possible source of error is that we have used a full conditional which may not be consistent with a valid prior distribution. However, we argue that this is probably not the cause of the errors in the realisations produced by the recursive algorithm: the Gibbs sampler used exactly the same prior full conditional and did not produce realisations with such poor definition of the channels' edges. It should be noted that although the factorisation condition was not satisfied, the positivity condition was. Thus the chain induced in the Gibbs sampling algorithm was certainly, at least theoretically, recurrent if it had continued to an infinite number of samples.

In summary, the results obtained using the new sampling algorithm seem good and robust, and we argue that the approximation errors discussed above appear at least no worse than the errors associated with the results of Gibbs sampling. Neither errors can be quantified. With Gibbs sampling we are consoled by the fact that in

the infinite limit the ensemble of samples will converge to the desired distribution, but for practical finite chains of samples this may never be the case.

In addition to attempting to estimate the approximation error a-priori, future work on the recursive algorithm should concentrate on its practical application to 3-D problems. We have discussed briefly how the recursive algorithm, and the sub-grid approximation, may be applied to 3-D grids. However, it is concerning that the number of floating point operations required increases exponentially with the approximation parameter (i.e., b) in the third dimension. We propose that, for practical application to 3-D grids, a different approximation scheme should be developed which further reduces the number of floating point operations required. The fundamental control on the computational expense of algorithm 2 is the size of the sample space of $\mathbf{g}$ (i.e., $\mathcal{G}^M$). This sample space is not explicitly chosen, but forced upon us by the choice of spatial parametrisation as a grid. It may not be optimal if a large part of the model space can be disregarded as a geological impossibility. This is often the case given the spatially structured nature of naturally occurring geology. If we call this segment of the model space - which may be assigned zero probability - $\mathcal{N}$ then the effective size of the model space should be $\mathcal{G}' = \mathcal{G}^M - \mathcal{N}$. It is clear that if we were able to somehow run algorithm 2 on $\mathcal{G}'$ rather than $\mathcal{G}^M$ then significant efficiency savings could be made. However, we found that implementation of this in practice is difficult since the division in equation 4.10 must be carried out with different irregular sample spaces for the denominator and numerator. Further work must be carried out before this approximation can be used effectively.

Extension of the algorithm to continuous variables g_i (such as those inverted for by Shahraeeni et al. (2012)) may be possible. However, it is likely that a sparse parametrisation of both the prior and likelihood (e.g., a Gaussian mixture model) would need to be chosen such that the computational cost may be controlled.

There are similarities between our recursive algorithm technique and multi-point geostatistical simulation techniques (Remy et al., 2009, pp.69-73). These techniques can be interpreted as trying to determine *a-priori* the partial conditionals $p(g_i|\mathbf{e}, \mathbf{g}_{<i})$ (Strebelle, 2002). This means that training images are produced of the $\mathbf{g}$ and corresponding $\mathbf{e}$ variables. Then the partial conditionals are determined empirically from these by using either machine learning techniques (Caers, 2001) or parametric estimation (Strebelle, 2002). The advantage of this approach is that, in theory, no computation is required to obtain the partial conditionals: they are simply learnt from 'examples' of $[\mathbf{g}, \mathbf{e}]$ and are ready for use immediately. In reality these exam-

ples are created by first generating a training image for the geological variable $\mathbf{g}$ and then using forward modelling to obtain $\mathbf{e}$. This is a significant computational burden, especially if the data generated by $p(\mathbf{e}_i|g_i)$ has high variance (as in the example presented in section 4.7). Indeed it may not even be possible to obtain enough samples in finite time, or with finite resources, to determine these partial conditionals sufficiently well. Consequently the $\mathbf{e}$ variable is often referred to as 'soft data' in such inversions, implying that it only constrains g_i locally e.g., $\mathbf{e}_i$ may only constrain g_i (Zhang et al., 2008). As in these geostatistical learning strategies, the recursive algorithm requires a training image of $\mathbf{g}$ to be generated so that the prior full conditional can be determined. However, a corresponding training image for $\mathbf{e}$ is not required: the recursive algorithm *analytically* incorporates the observed data into the computation of the partial conditionals using the cell-wise likelihood distributions. Thus the recursive algorithm may be a useful alternative to current geostatistical learning-based strategies.

We have also shown that the prior replacement operation developed in Chapter 3 can be used within stochastic geological inversion. Equation 4.9 in the recursive algorithm and equation 4.18 in Gibbs sampling both represent the application of a so-called new prior (i.e., the full conditional) to a likelihood distribution $p(\mathbf{e}_i|g_i)$. Such distributions can be determined from the results of neural network inversion, that is $\mathbf{e}_i \rightarrow p(g_i|\mathbf{e}_i)$ for all i, by removing the old prior used for training. Thus we have shown that the requirement that $p(\mathbf{g}) = \prod_{i=1}^{M} p(g_i)$ can be relaxed, and thus neural network inversion can be a useful method for stochastic geological inversion in general.

4.10 Summary

We have shown that the posterior distribution for spatial inverse problems can be sampled from exactly, by using a recursive algorithm to decompose that distribution as a set of conditional probability distributions which may be sampled from sequentially. However, this can only be achieved if the problem is specified by a grid of model parameters with coincident, independent likelihood information, and spatially correlated prior information specified using a full conditional distribution (i.e., if the local prior and likelihood properties are assumed). We have developed approximations to the recursive algorithm such that it may be applied efficiently to a

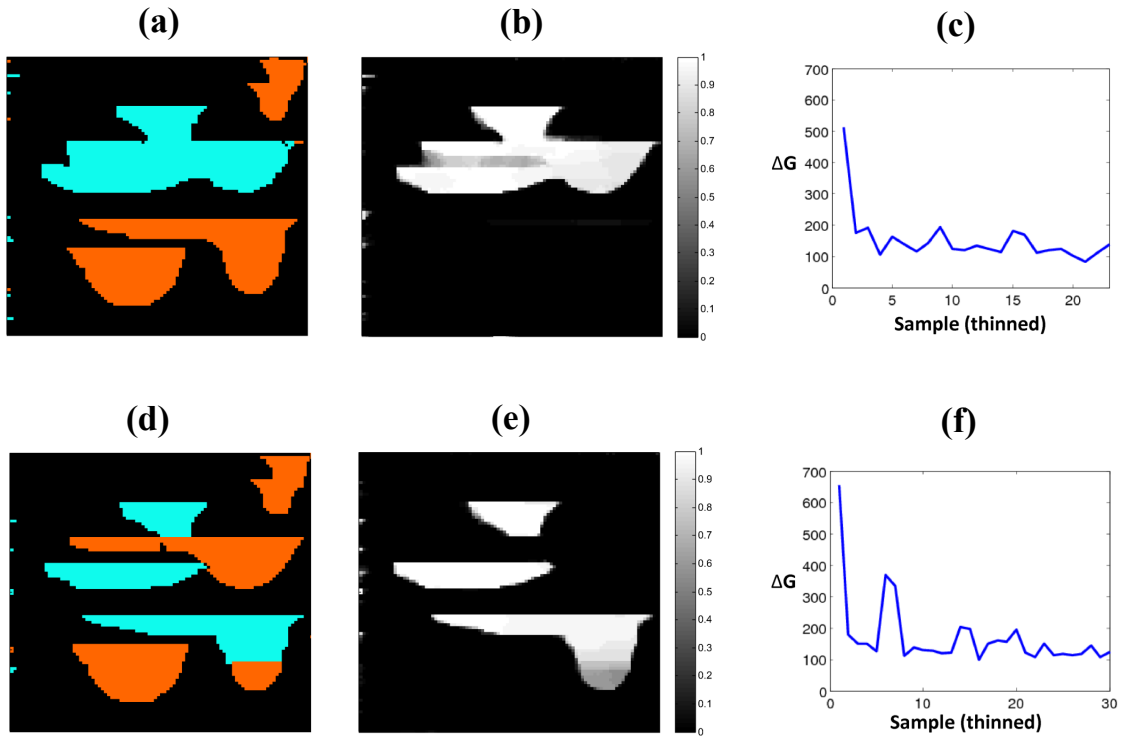

Figure 4.10: The results of running the Gibbs sampling algorithm (a Markov-chain Monte-Carlo method) to sample from the posterior $p(\mathbf{g}|\mathbf{e})$, with two different starting realisations shown in the upper and lower rows. The left column shows the final realisation after 1×10^8 iterations, the centre column shows the posterior cell-wise marginal probability of gas-sand occurrence (i.e., $p(g_i = \text{gas-sand}|\mathbf{e})$ at each cell), and the right column shows the total number of changes in facies between consecutive retained (post-thinning) realisations.

large 2-D grid of data. Because the posterior can be sampled from exactly, the well-known convergence problems of Markov-chain Monte-Carlo sampling algorithms are avoided. These algorithms (such as the Metropolis algorithm or Gibbs sampler) may not produce a set of samples which converge to the posterior (target) distribution in a practical time period.

We successfully applied the recursive algorithm to a synthetic geological inversion problem: we inverted seismic impedance data for lithology-fluid class. The synthetic data comprised noisy S- and P-wave impedances estimated at each cell in a 2-D grid. A training image was used to determine a suitable prior defined using a full conditional. From these two elements we estimated the posterior probability for the distribution of brine-sand, shale and gas-sand throughout the grid. The results of the recursive algorithm compared well to the results of Gibbs sampling on the same synthetic data. The results of Gibbs sampling showed significant bias: the use of such results would likely have led to one very significant gas-accumulation being completely unidentified. Both gas accumulations are reliably identified by the new recursive algorithm.

Thus the aim of developing a methodology for exact sampling from the geological posterior, which avoids bias, was achieved. Additionally, we also used prior replacement within the derivation of the new recursive algorithm and in Gibbs sampling. Thus, by extension, we showed that neural network inversion (with the addition of the prior replacement operation) can be useful in the context of general stochastic geological inversion where the geological prior is joint over $\mathcal{G}$ (that is, it is not defined as $p(\mathbf{g}) = \prod_{i=1}^{M} p(g_i)$).

References

Arnesen, P. (2010), Approximate recursive calculations of discrete Markov random fields, Ph.D. thesis, Norwegian University of Science and Technology.

Avseth, P., T. Mukerji, and G. Mavko (2005), *Quantitative seismic interpretation*, Cambridge University Press.

Bartolucci, F., and J. Besag (2002), A recursive algorithm for Markov random fields, *Biometrika*, *89*(3), 724–730.

Baum, L. E., T. Petrie, G. Soules, and N. Weiss (1970), A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, *The annals of mathematical statistics*, *41*(1), 164–171.

Belisle, C. (1998), Slow convergence of the Gibbs sampler, *Canadian Journal of Statistics*, *26*(4), 629–641.

Besag, J. (1974), Spatial interaction and the statistical analysis of lattice systems, *Journal of the Royal Statistical Society. Series B (Methodological)*, *36*(2), 192–236.

Besag, J., and P. J. Green (1993), Spatial statistics and Bayesian computation, *Journal of the Royal Statistical Society. Series B (Methodological),* pp.25–37.

Brook, D. (1964), On the distinction between the conditional probability and the joint probability approaches in the specification of nearest-neighbour systems, *Biometrika*, *51*(3/4), 481–483.

Brooks, S. P., and A. Gelman (1998), General methods for monitoring convergence of iterative simulations, *Journal of computational and graphical statistics*, *7*(4), 434–455.

Caers, J. (2001), Geostatistical reservoir modelling using statistical pattern recognition, *Journal of Petroleum Science and Engineering*, *29*(3), 177–188.

Caers, J. (2005), *Petroleum geostatistics*, Richardson, TX: Society of Petroleum Engineers.

Chen, L., Z. Qin, and J. S. Liu (2001), Exploring hybrid monte carlo in Bayesian computation, *Sigma*, *2*, 2–5.

Friel, N., and H. Rue (2007), Recursive computing and simulation-free inference for general factorizable models, *Biometrika*, *94*(3), 661–672.

Friel, N., A. Pettitt, R. Reeves, and E. Wit (2009), Bayesian inference in hidden Markov random fields for binary data defined on large lattices, *Journal of Computational and Graphical Statistics*, *18*(2).

Geman, S., and D. Geman (1993), Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, *Journal of Applied Statistics*, *20*(5-6), 25–62.

Gilks, W. R., S. Richardson, and D. J. Spiegelhalter (1996), *Markov chain Monte Carlo in practice*, CRC press.

Haario, H., E. Saksman, and J. Tamminen (1999), Adaptive proposal distribution for random walk Metropolis algorithm, *Computational Statistics*, *14*(3), 375–396.

Hammersley, J. M., and P. Clifford (1971), Markov fields on finite graphs and lattices. 1971, *Unpublished manuscript*.

Hastings, W. K. (1970), Monte Carlo sampling methods using Markov chains and their applications, *Biometrika*, *57*(1), 97–109.

Journel, A., R. Gundeso, E. Gringarten, and T. Yao (1998), Stochastic modelling of a fluvial reservoir: a comparative review of algorithms, *Journal of Petroleum Science and Engineering*, *21*(1), 95–121.

Kass, R. E., B. P. Carlin, A. Gelman, and R. M. Neal (1998), Markov chain monte carlo in practice: A roundtable discussion, *The American Statistician*, *52*(2), 93–100.

Kirkpatrick, S., D. G. Jr., and M. P. Vecchi (1983), Optimization by simmulated annealing, *science*, *220*(4598), 671–680.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953), Equation of state calculations by fast computing machines, *The journal of chemical physics*, *21*, 1087.

Mosegaard, K., and M. Sambridge (2002), Monte Carlo analysis of inverse problems, *Inverse Problems*, *18*(3), R29.

Olea, R. (1999), *Geostatistics for engineers and earth scientists*, Kluwer Academic Boston.

Remy, N., A. Boucher, and J. Wu (2009), *Applied geostatistics with SGeMS: a user's guide*, Cambridge University Press.

Rimstad, K., and H. Omre (2010), Impact of rock physics depth trends and Markov random fields on hierarchical Bayesian lithology fluid prediction, *Geophysics*, *75*(4), R93–R108.

Robert, C. P., and G. Casella (2004), *Monte Carlo statistical methods*vol.319, Springer, New York.

Rue, H., and L. Held (2005), *Gaussian Markov random fields: theory and applications*, Chapman & Hall.

Saul, L. K., and S. T. Roweis (2003), Think globally, fit locally: unsupervised learning of low dimensional manifolds, *The Journal of Machine Learning Research*, *4*, 119–155.

Scott, S. L. (2002), Bayesian methods for hidden Markov models, *Journal of the American Statistical Association*, *97*(457).

Shahraeeni, M. S., A. Curtis, and G. Chao (2012), Fast probabilistic petrophysical mapping of reservoirs from 3D seismic data, *Geophysics*, *77*(3), O1–O19.

Strebelle, S. (2002), Conditional simulation of complex geological structures using multiple-point statistics, *Mathematical Geology*, *34*(1), 1–21.

Tarantola, A. (2002), *Inverse problem theory: Methods for data fitting and model parameter estimation*, Elsevier Science.

Tjelmeland, H., and H. M. Austad (2012), Exact and approximate recursive calculations for binary Markov random fields defined on graphs, *Journal of Computational and Graphical Statistics*, *21*(3), 758–780.

Ulvmoen, M., and H. Hammer (2010), Bayesian lithology fluid inversion comparison of two algorithms, *Computational Geosciences*, *14*(2), 357–367.

Van Dyk, D. A., and T. Park (2008), Partially collapsed Gibbs samplers: Theory and methods, *Journal of the American Statistical Association*, *103*(482), 790–796.

Varma, M., and A. Zisserman (2003), Texture classification: Are filter banks necessary?, in *IEEE computer society conference on computer vision and pattern recognition, 2003. Proceedings.,* pp.II–691, IEEE.

Zhang, T., D. Lu, and D. Li (2008), A statistical information reconstruction method of images based on multiple-point geostatistics integrating soft data with hard data, in *International Symposium on Computer Science and Computational Technology, 2008. ISCSCT'08.,* pp.573–578, IEEE.

Chapter 5

Expert elicitation of the geological prior

5.1 Overview

In section 1.5.2 we described how the geological prior distribution $p(\mathbf{g})$ could be defined by a geostatistical model, and how the statistics of such a model could be obtained from training images. However, as was explained there, suitable real training images often do not exist for this purpose. An alternative is to generate training images based on expert knowledge, but this can be costly and inaccurate. In this chapter we describe a *general* method for obtaining the statistics of a geostatistical model (such as the full conditional probabilities in equation 1.13) directly from an expert. We demonstrate the methodology for a geostatistical model of a rock at pore-scale, but since the method is general it may be immediately applied to obtain statistics which specify the geological prior distribution $p(\mathbf{g})$ used in seismic inversion.

5.2 Introduction

In many geological disciplines geostatistical models are used to model the spatial relationships between geological features of interest within a certain area or volume of the subsurface (henceforth referred to as the *target geology* in this chapter). Such a model can be used to generate stochastic realisations of the target geology (Journel et al., 1998). For example, geostatistical models may be used to create realisations

of the distribution of pores within a rock or soil, and such realisations may be used to simulate flow in a subsurface reservoir (e.g., Keehm et al., 2004; Okabe and Blunt, 2005; Wu et al., 2006). Or, alternatively, they might be used to describe lithology distributions for estimating expected ore reserves in mining applications (Matheron, 1963; David and Blais, 1977; Dimitrakopoulos, 1998). Or, in the case of seismic inversion, they are used to specify a prior probability distribution over some set of geological parameters.

Such models require calibration statistics (controlling parameters) that are appropriate for each application; we refer to these as the *ideal statistics* in this chapter. Ideal statistics can be determined from the analysis of analogue geological formations. Photographs (Dueholm and Olsen, 1993), core samples (Zhang et al., 2009) or even geophysical survey results (Caers et al., 1999) from analogue formations may directly provide training images from which the ideal statistics can be extracted (Pringle et al., 2004, 2006; Price et al., 2008), but their relevance depends on the true similarity of the analogue and target formations (Ringrose et al., 1999; Kupfersberger and Deutsch, 1999; Truong et al., 2013).

It is widely accepted that a lack of suitable analogue formation data is a significant problem in geostatistics (Cui et al., 1995; Kerry and Oliver, 2007; Truong et al., 2013). Consequently, subjective information on the ideal statistics, obtained from geological experts via a process of elicitation, is increasingly incorporated within such analyses (Curtis, 2012). Using this approach, statistics which generate realisations consistent with the experts' mental envisagement of the target geology must be elicited. In the past this has been achieved by either:

1. creating realisations using the geostatistical model with a range of different statistics until an image which corresponds to their envisagement of the target geology is produced (e.g., Caers, 2005, pp. 18-26), or

2. producing a training image manually (Honarkhah and Caers, 2010; Comunian et al., 2011) or from geological process models (Nordahl et al., 2005) from which ideal statistics may be calculated.

Using approach (i), if the geostatistical model is appropriate to the application then after a sufficient number of iterations the expert may find a realisation which matches their envisagement of the properties of the target geology. They can take the statistics of that model to be an estimate of the ideal statistics. However, the number of

iterations required to reach that point may be very large - well beyond the fatigue limit of the expert(s). Ideally approach (ii) will automatically result in ideal statistics because the expert produces an appropriate training image of the target geology (Michael et al., 2010). However producing a training image is clearly subjective as it is highly unlikely that two people would produce identical images, or even use the same geological concepts to describe a particular scenario (Bond et al., 2007, 2012). Geological process models also require subjective choices to be made about which processes to include and which values to use for process-controlling parameters (Wood and Curtis, 2004; Hill et al., 2009). Additionally, using any of these approaches we are first obliged to choose a geostatistical model for which to find the statistics (parameters); this model is always wrong - it is a necessary simplification of reality (Leuangthong et al., 2004). Loquin and Dubois (2010) provide a detailed discussion of the resulting errors. In practice such epistemic error may be counteracted by modification of the statistics away from the strictly numerically best-fitting values obtained from the training image, but this again requires subjective judgements to be made. Both methods (i) and (ii) therefore have the potential to be very costly in terms of expert time and associated computation, and both are in part subjective.

Interrogation techniques designed to obtain robust quantitative estimates of the knowledge and uncertainty of individual and groups of experts have been developed in the field of expert elicitation (Tversky and Kahneman, 1974; Lindley et al., 1979; Kynn, 2008; James et al., 2010). Such techniques have been used successfully to obtain probability distributions over geological parameters (Lindley, 1983; Baddeley et al., 2004; Curtis and Wood, 2004). However, they have not been applied widely to the estimation of parameters of geostatistical models in particular. An exception is the recent work of Truong et al. (2013) who used formal elicitation techniques (e.g., Knol et al., 2010) to obtain estimates of the parameters of a variogram: they asked a group of experts to complete a set of on-line questions about the *numerical* values of the ideal statistics for a variogram model of earth surface temperature variability. They then pooled the opinions of the individual experts, using the formal rules of elicitation, to obtain an estimate (including uncertainty) of the ideal statistics. Truong and Heuvelink (2013) used a similar approach to estimate the parameters of a variogram describing the error on soil maps. The disadvantage of such an approach, which uses numerical information, is that it requires the expert to have some knowledge of the mathematics of the underlying model. This might not be appropriate for a geological expert who works mainly with visual data, but who

nevertheless has good intuition about the likely spatial relationships of geological objects (i.e., variables).

We propose an alternative method for obtaining ideal statistics directly from an expert without costly intermediate steps, and without requiring the expert to understand the mathematics or statistics of the underlying model. This methodology combines the principles of elicitation (Baddeley et al., 2004; Curtis and Wood, 2004) with recent advances in so-called 'interactive inversion' (Boschetti and Moresi, 2000, 2001) in which a genetic algorithm is used to constrain an inversion with the input of a geological expert. By contrast to the methods of Wood and Curtis (2004) and Truong et al. (2013), our approach does not require any numerical input to be given by the expert. A geological expert can therefore focus on their own area of expertise - analysing spatial (geological) patterns.

In this chapter, after briefly discussing notation in section 5.3, we describe our elicitation methodology in more detail and explain how the use of a genetic algorithm (GA) is key to its efficiency in sections 5.4 and 5.5. In section 5.6 we then describe an application to constrain the statistics of a particular multi-point geostatistical model which has been used in the past to model pore-spaces in reservoir rocks (Wu et al., 2006) and soils (Wu et al., 2004), as well as subsurface facies distributions (Stien and Kolbjørnsen, 2011). We demonstrate the effectiveness of the methodology by showing that ideal statistics can be estimated efficiently for this model, but also show how the method can be used to assess the uncertainty associated with the geological expert's judgement when using the algorithm.

5.3 Notation

The notation used in this chapter follows that used in the introduction. We will demonstrate the elicitation algorithm for the elicitation of discrete geological parameters only. Thus, $\mathbf{g}$ is used to represent the geological parameters here (although the method can be easily generalised to continuous geological parameters $\mathbf{m}$). A summary of the notation used in this chapter is given in Appendix H.4.

5.4 Elicitation methodology

Suppose that we have a probability distribution $p(\mathbf{g})$ over the geological parameters $\mathbf{g}$ in a (1-D, 2-D, or 3-D) grid. We assume that $p(\mathbf{g})$ is defined by a geostatistical model, parametrised by a vector of statistics $\mathbf{T}$ which we can write as $\mathbf{T} = \{t_k \mid k \in \{1, 2, ..., L\}\}$ where L is the number of elements in the vector (or number of required statistics). Thus (in principle, at least) we can make a realisation

$$\mathbf{g} \sim p(\mathbf{g}|\mathbf{T}), \tag{5.1}$$

where we have explicitly noted a dependence of the distribution on some given vector $\mathbf{T}$. As discussed in section 1.5.2 the geostatistical model will either be two-point or multi-point in nature. In the former case, $\mathbf{T}$ might correspond to the parameters describing a variogram or to the parameters of a Gaussian distribution (in the non-parametric and parametric approaches, respectively). For a multi-point model, $\mathbf{T}$ might correspond to the probabilities within the full conditional distribution (equation 1.13). In principle, the choice of geostatistical model makes no difference to the algorithm presented here, as long as it permits sampling to be performed as in equation 5.1

Our elicitation methodology obtains an estimate of the ideal statistics, $\mathbf{T}_{best}$, directly from the expert. Or in other words, it estimates the statistics which induce $p(\mathbf{g})$ to produce realisations of the geology which are consistent with the expert's envisagement of the target geology, for a given application. This involves iteratively improving a small population of candidate statistics vectors $\mathcal{S} = \{\mathbf{T}_1, ..., \mathbf{T}_j, ..., \mathbf{T}_P\}$, where P is the number of statistics vectors in the population. Using $p(\mathbf{g}|\mathbf{T})$, each member of this population can be used to generate a realisation: thus we obtain an associated set of realisations $\mathcal{R} = \{\mathbf{g}_1,...,\mathbf{g}_j,...,\mathbf{g}_P\}$ where $\mathbf{g}_j \sim p(\mathbf{g}|\mathbf{T}_j)$. Note that the index j will be used consistently in this chapter to reference members of a population; it should not be confused with the index i which we will consistently use to reference the individual geological parameters at each cell in the grid (i.e., $\mathbf{g} = [g_1, ..., g_i, ..., g_M]$).

In each iteration of our method, $\mathcal{S}$ is updated using three genetic algorithm operations (which are similar to evolutionary processes in nature), the details of which are given in the next section and are controlled by the fitness of each $\mathbf{T}_j$ in $\mathcal{S}$ with respect to some criterion. Here, the criterion for the fitness of $\mathbf{T}_j$ is how well

its corresponding realisation $\mathbf{g}_j$ in $\mathcal{R}$ matches the target geology. The target geology is not physically accessible, as it is envisaged only mentally by the expert. We therefore ask the expert's opinion on how well each $\mathbf{g}_j$ matches their envisagement of the target geology (the *fitness* of $\mathbf{g}_j$). The GA operations only require a relative ranking between the members of $\mathcal{S}$ (Goldberg, 1989) and thus the expert is only asked to rank the set of $\mathbf{g}_j$ variables according to their relative fitness within $\mathcal{R}$. The algorithm continues to iterate until a $\mathbf{g}_j$ is found which adequately matches the expert's envisagement of the target geology, in their opinion. The corresponding $\mathbf{T}_j$ is then retained as the ideal statistics vector, $\mathbf{T}_{best}$.

To summarise, using l to denote iteration number, the algorithm begins at $l = 0$ with a randomly generated initial population of statistics vectors $\mathcal{S}_{l=0}$, and then proceeds as follows:

1. Use $\mathcal{S}_l = \{\mathbf{T}_1, ..., \mathbf{T}_j, ..., \mathbf{T}_P\}$ in $p(\mathbf{g}|\mathbf{T}_j)$ to generate a set of realisations $\mathcal{R}_l = \{\mathbf{g}_1,...,\mathbf{g}_j,...,\mathbf{g}_P\}$.

2. Display the set of realisations in $\mathcal{R}_l$ to the expert or experts.

3. Ask the expert(s) to rank (from 1 to P, with 1 being the best ranking) each $\mathbf{g}_j$ in $\mathcal{R}_l$. Associate ranking of each $\mathbf{g}_j$ to the corresponding statistics vector $\mathbf{T}_j$ in $\mathcal{S}_l$.

4. If the expert(s) decide that one of the realisations ($\mathbf{g}_j$) is adequately consistent with their mental envisagement of the target geology then stop, retaining the corresponding statistics as $\mathbf{T}_{best} = \mathbf{T}_j$. If not continue to step (5).

5. Apply genetic algorithm (GA) operations to the ranked set $\mathcal{S}_l$, yielding a new population of statistics vectors, $\mathcal{S}_{l+1}$.

6. Set $l = l + 1$. Return to step (1).

The ranking in step (iii) does not need to be made over each member of $\mathcal{R}_l$. That is to say that we can specify (or the expert could choose) that the expert need only rank P^* of the realisations where $P^* \leqslant P$, with the ranking running from 1 to P^*. In this case any unranked members of $\mathcal{S}_l$ are discarded and play no part in the generation of $\mathcal{S}_{l+1}$. This will be useful later.

Of course, we may be concerned about the representativeness of any single realisation $\mathbf{g}_j$ of the corresponding statistics, $\mathbf{T}_j$, since it is generated randomly. A

simple solution is to present realisations which are as large as possible (in terms of the number of cells in the grid realised) such that the probability of displaying a realisation with the desired statistical properties (i.e., those specified by $\mathbf{T}_j$) is maximised. Alternatively, multiple realisations for a single $\mathbf{T}_j$ could be made in step (ii) and an average ranking could be obtained. We aim to keep the algorithm as simple as possible so we use the former strategy in our implementation of the algorithm but the latter would be equally valid. In the next section we explain in detail the genetic algorithm operations applied to $\mathcal{S}_l$, and how on average they improve the population (with respect to the criterion described above).

5.5 Genetic algorithm operations

Technically speaking the procedure described above, of iteratively improving a population in order to find optimal parameter values (in this case a vector of statistics), constitutes a genetic algorithm given the appropriate choice of operations applied to the ranked $\mathcal{S}_l$ population in order to form the new population $\mathcal{S}_{l+1}$ (Goldberg, 1989). In order of application these operations are:

1. **Reproduction** In this step a new set of P reproductions are made of the statistics vectors in $\mathcal{S}_t$. An element of $\mathcal{S}_t$ is chosen to be reproduced randomly with probability inversely proportional to their ranking (i.e., the better ranked the parameter vector, the more probable it is that it will be reproduced). The resultant set of the first P new parameter vectors reproduced is denoted $\mathcal{S}'_t$. Note that when $P^* < P$ any unranked population members are assigned zero probability of reproduction and hence play no further part in the generation of the new population.

2. **Mating and crossover** The members of $\mathcal{S}'_t$ are randomly paired (or 'mated'). Each pair of vectors then swaps a randomly determined number of their elements, producing the next stage of the population, $\mathcal{S}''_t$.

3. **Mutation** Each element of each vector in $\mathcal{S}''_t$ may be perturbed randomly. The probability that a given element is perturbed is given by the parameter β, and the magnitude of perturbation is controlled by the parameter α. The exact form of the mutation operation is application-specific: it is dependent upon

the domain of the statistics vector $\mathbf{T}$. This completes the genetic algorithm operations, and produces the new population, $\mathcal{S}_{l+1}$.

The analogy between the natural processes of genetic evolution and these operations is clear from their names. It is also clear why, on average, they might be expected to improve the population with respect to the expert's opinion, yet retain diversity within $\mathcal{S}_t$: the 'Reproduction' step ensures that the best members of the population are retained. The 'Mating and crossover' step interchanges and splices the 'genes' of these already good individuals in the hope that the next population will contain improved individuals. The 'Mutation' step introduces some random perturbation to the 'gene pool' so as to ensure mobility around the parameter space (any good new mutations are more likely to survive subsequent iterations, as bad mutations are likely to be removed by the 'Reproduction' step). The random nature of the genetic operators is important as in theory this prevents the algorithm from becoming stuck in local minima, thus the space of possible $\mathbf{T}$ vectors is better explored (Goldberg, 1989).

The GA differs from optimisation techniques in a number of other ways. The most important of these for our application is that absolute values for the fitness of the statistics vectors are not required: only their relative ranking is required. This is important because obtaining meaningful absolute fitness values from the expert would be virtually impossible. Furthermore, fitness gradients with respect to changes in the vectors $\mathbf{T}_j$, are not required (as is the case for many linearised optimisation methods). Gradients could potentially be obtained from the expert but they would be very time consuming to elicit, even for a single point in the space of possible $\mathbf{T}$ vectors.

Clearly, there are a number of algorithmic parameters within the genetic algorithm operations, such as the mutation parameters (α and β), or the proportionality between rank in $\mathcal{S}_t$ and probability of reproduction, that we have not defined explicitly. These parameters effect the way the algorithm explores the space of statistic vectors (henceforth the *dynamics* of the algorithm) and therefore the convergence rate of the algorithm. We found that in the application below it took little effort to determine reasonable values for these parameters through a process of trial and error, which permitted convergence within an acceptable number of iterations. Thus for brevity we will not discuss these parameters further and such parameter values are kept constant for all results presented here, with the exception of the mutation

parameters α and β, which we will vary later. These parameters define the maximum size and number of perturbations applied to the statistics vector $\mathbf{T}_j$. They are therefore important in determining the dynamics of the algorithm because they control the (expected) step-size that the algorithm uses to explore the space of statistics vectors.

5.6 Example application to pore-space modelling

As stated above, the method of direct elicitation can be applied to any geostatistical model including those used to define $p(\mathbf{g})$ used in seismic inversion. We will actually demonstrate the elicitation methodology for estimating the statistics which parametrise a geostatistical model where the geological parameters $\mathbf{g}$ describe the distribution of a rock's pore-space. However, a cellular grid is used to model the pore-space, which is identical to the grids used previously to model the geological parameters for seismic inversion, the only difference is the scale of the cells. Thus we will demonstrate that the method is directly applicable to determining $p(\mathbf{g})$ for seismic inversion. We will first describe the geostatistical model, and then explain how we use the GA method in practice to estimate the ideal statistics (in the opinion of individual experts) to represent specific target pore-space topologies. Finally we describe how we demonstrate the algorithm's performance in practice by allowing 12 experts to use the algorithm to determine ideal statistics.

5.6.1 Pore space modelling

We use a 2-D binary image model which contains two materials 'pore' and 'matrix', and a multi-point geostatistical model, $p(\mathbf{g}|\mathbf{T})$, to represent the spatial dependency between these two materials (Wu et al., 2004, 2006; Stien and Kolbjørnsen, 2011). The image is modelled using a 2-D cellular grid identical to the subsurface model grids used previously for seismic inversion: the grid has M cells with the usual $x \in [1, ..., X]$, $z \in [1, ..., Z]$ coordinate system, and indexing as shown in Figure 5.1(a)). Each cell is associated with a binary variable $g_i \in \{pore, matrix\}$. Thus a geological parameter vector $\mathbf{g} = [g_1, ..., g_i, ..., g_M]$ as used in previous chapters (albeit at a different scale) can be used to describe the material in all cells. To define $p(\mathbf{g}|\mathbf{T})$ we begin, for the moment, by ignoring the statistics (parameters) of the distribution $\mathbf{T}$, and write the joint probability distribution $p(\mathbf{g}|\mathbf{T})$ as a product of individual

conditional probability distributions over g_i given $\mathbf{g}_{<i} = [g_1, ..., g_{i-1}]$ (that is all the variables in cells previous to i in the indexing of the grid as shown in Figure 5.1(b)):

$$p(\mathbf{g}|\mathbf{T}) = \prod_M p(g_i|\mathbf{g}_{<i}, \mathbf{T}). \tag{5.2}$$

Now, before defining $\mathbf{T}$, we make two further simplifications to this distribution. Firstly we specify that the variable g_i is conditionally independent of most of the variables in $\mathbf{g}_{<i}$. That is, g_i is only dependent on a smaller subset of them, the neighbourhood $Ne(i)$, of cell i. Thus,

$$p(\mathbf{g}|\mathbf{T}) = \prod_M p(g_i|\mathbf{g}_{Ne(i)}, \mathbf{T}), \tag{5.3}$$

where $Ne(i)$ is defined as a subset of indices 'previous to' cell i which define the neighbourhood (note the definition of the neighbourhood used here is more specific compared to the more general definition in equations 1.13 and 4.1). The neighbourhood is typically (but not necessarily) a set of adjacent cells, thus it is defined as a function of cell i. For example the neighbourhood in Figure 5.1(c) is written $Ne(i) = \{i-1, i-X-1, i-X\}$, where X is the lateral dimension of the grid. $|Ne(i)|$ is used to denote the number of the elements of $Ne(i)$. The second simplification is that the conditional distribution and (the shape of) the neighbourhood are invariant to the position in the grid, i.e., $Ne(i)$ and $p(g_i|\mathbf{g}_{Ne(i)})$ are invariant to i.

The statistics of the distribution, that is the $\mathbf{T}$ vector, can now be defined. Because of the invariance to position the only statistics required by the model are those describing this single, *general* conditional probability distribution $p(g_i|\mathbf{g}_{Ne(i)})$. Thus $\mathbf{T}$ need only specify these conditional probabilities for each possible pore-matrix configuration of the neighbouring cells, $\mathbf{g}_{Ne(i)}$. Furthermore, since $p(g_i = pore|\mathbf{g}_{Ne(i)}) = 1 - p(g_i = matrix|\mathbf{g}_{Ne(i)})$, it is sufficient for $\mathbf{T}$ to define just the probabilities $p(g_i = pore|\mathbf{g}_{N_i})$ in order that a valid probability distribution is specified. Specifically, we define $\mathbf{T}$ by first introducing $\mathcal{C}$ as the set of all possible configurations of $\mathbf{g}_{Ne(i)}$. Since g_i is binary the size of $\mathcal{C}$ is related to the size of the neighbourhood by $|\mathcal{C}| = 2^{|Ne(i)|}$. Then returning to the definition of the vector of statistics as $\mathbf{T} = \{t_k \mid k \in \{1, 2, ..., L\}\}$ each element is now a probability $t_k = p(g_i = pore|\mathbf{g}_{Ne(i)} = \mathcal{C}(k)) \in [0, 1]$. In words, each t_k element is the probability of cell i being pore given that the neighbourhood of i contains the k^{th} configuration of pore and matrix in $\mathcal{C}$. Consequently the size of the $\mathbf{T}$ vector is simply the size

of $\mathcal{C}$, i.e., $L = |\mathcal{C}| = 2^{|Ne(i)|}$. The index k will be used consistently in this chapter to reference members of the $\mathbf{T}$ vector (and thus is distinct from the i and j indices defined earlier in this chapter).

It should be noted that at the edges and corners of the grid, $Ne(i)$ and $p(g_i|\mathbf{g}_{Ne(i)})$, cannot be the same as those in the middle of the grid since there are 'missing' neighbours (i.e., the invariance to position does not apply here). However, appropriate modifications can be made to $\mathbf{T}$ to obtain appropriate conditional probabilities at such positions, and this makes no fundamental difference to the method.

The conditional probability distributions which comprise the decomposition of $p(\mathbf{g})$ (equations 5.2 and 5.3) are dependent only on previous cells in the indexing of the grid. Thus sampling from $p(\mathbf{g})$ can be performed exactly using sequential simulation (Stien and Kolbjørnsen, 2011), and hence the required realisations $\mathbf{g} \sim p(\mathbf{g}|\mathbf{T})$ can be generated efficiently for a given statistics vector $\mathbf{T}$. This means that realisations can be presented to the expert almost immediately, so there is no need for the expert to wait for realisations to be generated. This is not a requirement of the algorithm: there is no reason why the algorithm cannot pause for some time between the points where it requires the expert's input, as long as the total run time is reasonable. However, we have chosen a real geostatistical model which can generate realisations very rapidly such that the concept of the elicitation methodology can be proven quickly and conclusively.

It is clear that the geostatistical model presented here is a particular instance of the multi-point model defined by equation 4.1, with a certain non-symmetric neighbourhood structure $Ne(i)$. However, the choice of possible neighbours of i is restricted to a subset of the cells previous to cell i in the indexing system. This means that the 'full conditionals' in this case immediately permit exact, sequential sampling of $\mathbf{g}$ from $p(\mathbf{g})$ (which is in contrast to full conditionals in their general form).

In any case, the geostatistical model described above can be used to generate a set of realisations, $\mathcal{R} = \{\mathbf{g}_1,\mathbf{g}_2,...,\mathbf{g}_P\}$, given a population of P statistics vectors, $\mathcal{S} = \{\mathbf{T}_1, \mathbf{T}_2, ..., \mathbf{T}_P\}$. Thus, in theory we can use the elicitation methodology described in section 5.4 to find $\mathbf{T}_{best}$ (from step (iv) of the elicitation algorithm) for a given application. However, we must make some practical developments to the GA algorithm in order to do this, which we describe in section 5.6.2.

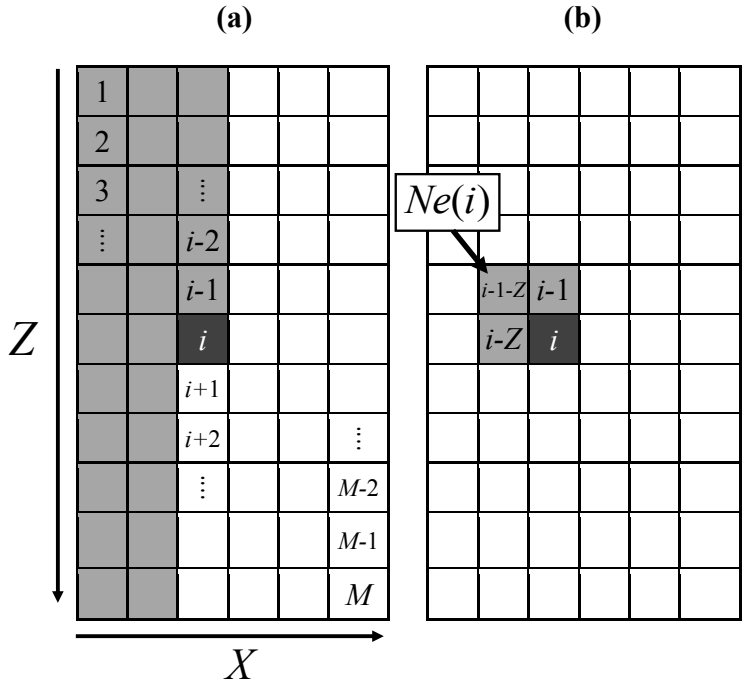

Figure 5.1: (a) Definition of an example 2-D grid and its indexing. (b) An illustration of a conditional distribution used in the decomposition of the probability distribution (the geostatistical model) in equation 5.2: the dark grey shaded cell contains the variate g_i and the light gray shaded cells are those containing conditioning variables. (c) The same conditional distribution but with dependencies limited to a restricted neighbourhood of cells. Again, the dark grey shaded cell contains the variate and the light gray shaded cells contain the conditioning neighbour variables in $Ne(i)$.

5.6.2 Practical application of the GA

The first practical consideration is the definition of the mutation operation for this application (and how this is related to the mutation parameters α and β); as explained above this is necessary since the statistics may have bounded domains. In our application the statistics vector $\mathbf{T}$ comprises a set of probabilities denoted t_k, which therefore have domain on the interval $[0, 1]$. In this case if a t_k element is chosen to be mutated (with probability equal to the mutation parameter β) then the new mutated value, t'_k, is randomly generated from the Uniform distribution

$$t'_k \sim \mathcal{U}\left[max\left(t_k - \alpha, 0\right), min\left(t_k + \alpha, 1\right)\right]. \tag{5.4}$$

where α is the mutation parameter (which controls the magnitude of mutation) and limits have been imposed at $\{0, 1\}$ to ensure that the mutated vector element is still a valid probability.

Perhaps the most important practical consideration is how the expert interacts with the GA. We designed a Graphical User Interface (GUI) which displays the members of the current population $\mathcal{S}_l$ to the expert, and which allows them to rank the realisations $\mathbf{g}$ in $\mathcal{R}_l$, and hence $\mathbf{T}$ in $\mathcal{S}_l$, using only mouse clicks. Empirically we have found that it is often difficult for an expert to start the GA (i.e., perform ranking) on an initial, random population since these tend to produce realisations which are highly non-geological. Thus we designed a two-stage algorithm with two implementations of the GUI, where the first stage was designed to obtain a good starting population for the algorithm. In this stage the population was relatively large with $P = 24$. The realisations presented to the expert were also relatively large in terms of the size of the grid simulated ($X = 120$ and $Z = 120$) but were displayed with relatively low magnification with 4.3 cells/mm; this configuration is intended to allow the expert to identify important large-scale statistical/geological features of the realisation rapidly. Furthermore, the α_1 and β_1 parameters (where the subscript 1 indicates that these parameters are used in the first stage of the algorithm only) were relatively large with both being ~ 0.4 (although they were allowed to vary slightly between experts - the reason for this will be explained later). The motivation for designing the first stage in this manner was to present the expert with a diverse population that evolves rapidly such that they may find a general area of the space of $\mathbf{T}$ vectors which provides realisations with geology consistent with

the target geology. Since there were a large number of realisations at this stage, the expert was only asked to rank the three best, i.e., $P^* = 3$. A typical screen-shot from the first stage GUI is shown in Figure 5.2.

The experts were asked to perform the ranking at each iteration, using the 'Next' button in the GUI to indicate that ranking was complete and that the algorithm could continue to the next iteration. The vectors $\mathbf{T}_{rank\leqslant 3}$ (obtained from $\mathbf{g}_{rank\leqslant 3}$) would then be passed to the GA operations in order to produce the population for the next iteration, $\mathcal{S}_{l+1}$. They were asked to continue using the GUI in this way until the current population contained a realisation which they thought had (statistically) the same geology as the target geology. They then ranked the population as usual but instead of pressing the 'Next' button they were instructed to press the 'Match' button in the GUI. At this point the second stage of the algorithm would begin (and the second stage of the GUI would be displayed). As explained above the output of the first stage is a starting population for the second stage: this population was made up of the three vectors $\mathbf{T}_{rank\leqslant 3}$ obtained from $\mathbf{g}_{rank\leqslant 3}$ at the last iteration of the first stage.

The second stage is designed to encourage the expert to look at the realisations in greater detail and to 'fine tune' the population of realisations in terms of their similarity to the target geology. Consequently, the population is much smaller with $P = 6$ and the size of the realisation grids is slightly smaller ($X = 90$ and $Z = 90$) than in the first stage; this permits the images to be magnified much more than in the first stage (with only 1.6 cells/mm). Furthermore, the α_2 and β_2 parameters (where the subscript 2 indicates that these parameters are used in the second stage only) were relatively small, both being ~ 0.15 (although again they were allowed to vary slightly between experts - the reason for this will be explained later).

As stated above, the first population of the second stage is derived from the top three ranked members of the population at the end of the first stage, $\mathbf{T}_{rank\leqslant 3}$. Since in the second stage $P = 6$, each of these three vectors must be replicated once to produce a total of six vectors (to become a valid first population for the second stage). Since there were fewer realisations to compare in the second stage the experts were asked to rank all six members of the population, i.e., $P^* = 6$. As in the first stage, the experts were asked to rank the realisations presented to them in the GUI at each iteration, using the 'Next' button to indicate that ranking was complete. They were asked to continue until the current population contained a realisation which they thought had (statistically) the same geology as the target geology. They then

ranked the population as usual but instead of pressing 'Next' they were instructed to press the 'Match' button in the GUI; once that button was pressed, the algorithm takes $\mathbf{g}_{rank=1}$ to be the realisation which the expert had found to match the target geology $\mathbf{g}_{best}$. Thus $\mathbf{T}_{rank=1}$ is taken to be the estimate of the ideal statistics $\mathbf{T}_{best}$. At this point the algorithm terminates. A typical screen-shot from the second stage GUI is shown in Figure 5.3.

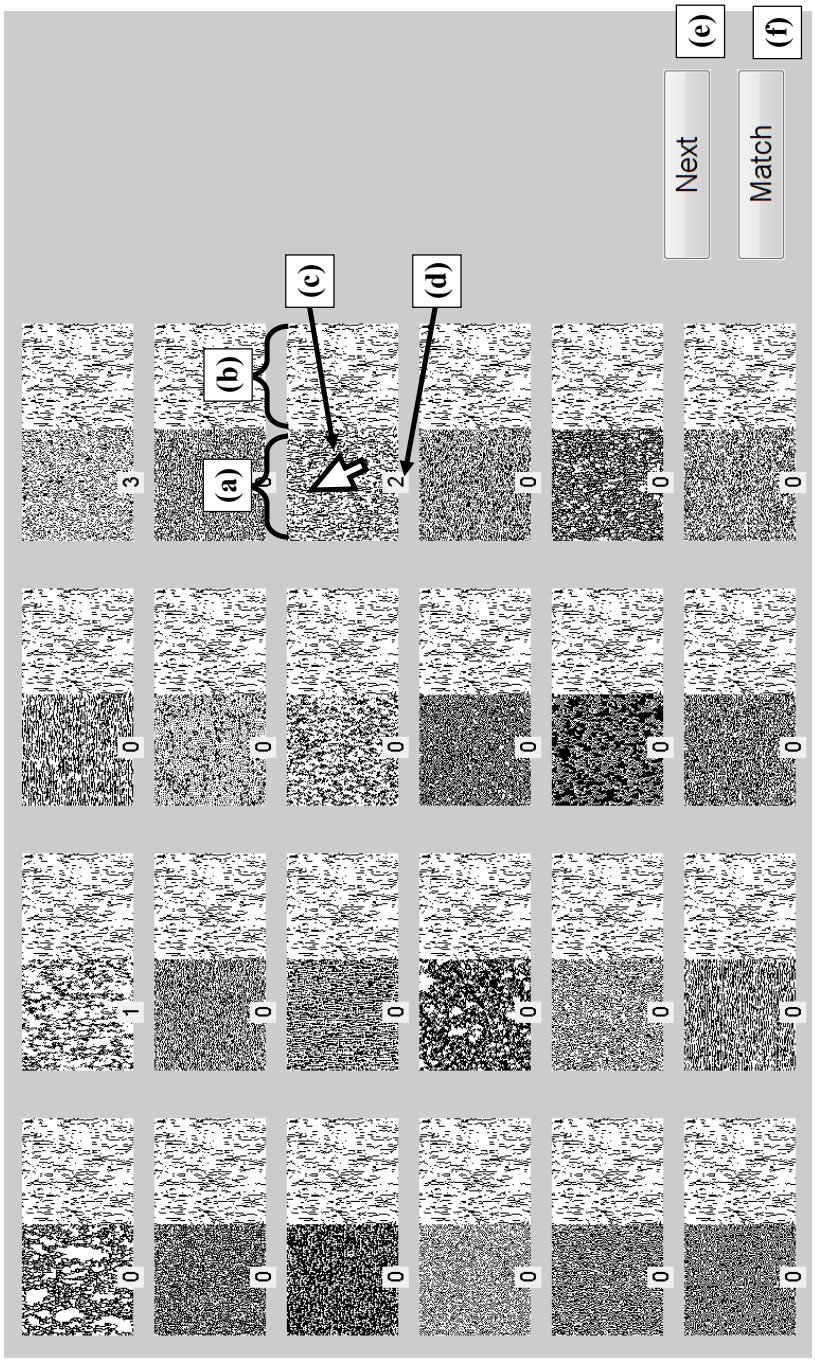

Figure 5.2: A screen-shot from the Graphical User Interface (GUI) used to implement the first stage of the algorithm. The expert is asked to rank 3 of the 24 pairs presented. Important elements of the GUI are indicated: (a) the left image in each pair is a realisation from the population; (b) the right image is the target pore-space image and is the same for all 24 realisations; (c) experts use the cursor to assign a ranking to the realisations by simply clicking the realisations in rank-order; (d) rankings selected are displayed beneath each realisation; (e) the expert presses the 'Next' button if they have finished ranking and wish to continue; (f) the expert presses the 'Match' button if they have found a realisation which adequately matches the target in their opinion.

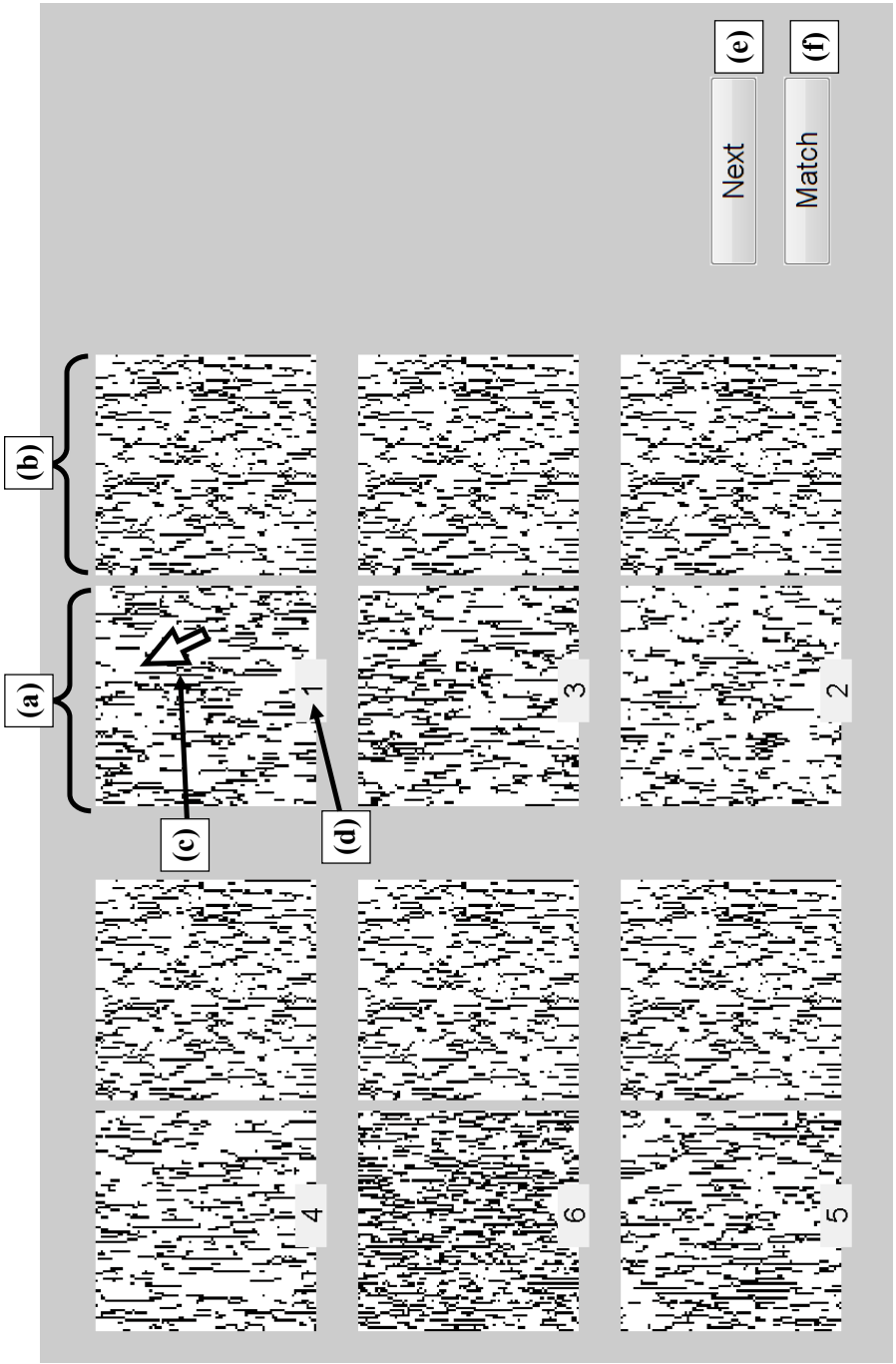

Figure 5.3: As for Figure 2, but for the second stage of the GUI. Here the expert is presented with 6 realisations and is asked to rank all 6 of these.

5.6.3 Testing the algorithm

In order to demonstrate the methodology we asked 12 geoscientists with varying backgrounds to use the elicitation methodology (via the GUI) to estimate ideal statistics $\mathbf{T}_{best}$ for a certain target pore-space geology. The geostatistical model was defined to have the neighbourhood as shown in Figure 5.1(c) (thus $|Ne(i)| = 3$ and $|\mathcal{C}| = 2^3 = 8$). Since we wanted to test the method in a controlled way, the target pore-space was actually provided to the subjects: that is, a pore-space image was displayed to them, and they were asked to use the pore distribution in that image as the target geology. Their goal was to find statistics that generated pore-space images with the same statistical distribution as that of the target image. Thus, given that we know the target image (and its statistics) in each of these tests, we were able to assess exactly how well the expert performed - which would not have been possible if they were matching a concept or image held only in their mind. Note that in a real application of the algorithm no physical target image would be presented to the expert; instead they would be asked to use their mental envisagement of the target geology for comparison.

The target image itself had been created using the same geostatistical model as used to create realisations in the algorithm above, and therefore had been created with an actual statistics vector, $\mathbf{T}_{target}$. In mathematical terms the target image was a realisation, $\mathbf{g}_{target} \sim p(\mathbf{g}|\mathbf{T}_{target})$. Importantly this allowed us to measure the numerical convergence rates towards $\mathbf{T}_{target}$ as the subjects used the GUI. We emphasise that none of the numerical information about the target statistics was used in the algorithm, nor was it provided to the subjects; it was only used for the purpose of assessing the performance of the experts. The only information used by the elicitation algorithm was provided through each expert's subjective ranking provided through the GUI. Two different target statistics vectors were used to generate two different target pore-space distributions for the experts. The first vector produced a so-called 'crack-pore' distribution: vertically aligned elongated pores with some isolated micro-porosity. The second produced a so-called 'round-pore' distribution: more rounded pores with much more micro-porosity within the matrix. The experts were divided into two groups of six. One group was provided with 'crack-pore' target images, the other with 'round-pore' target images.

Given that we knew $\mathbf{T}_{target}$ we could also test whether the expert was actually able to obtain a $\mathbf{T}_{best}$ vector which produces realisations $\mathbf{g}_{best}$ with geologies which

were truly indistinguishable, to the best of that expert's ability, from the target geology $\mathbf{g}_{target}$. This could be achieved after the expert pressed the 'Match' button by presenting them with a population comprising a mix of realisations generated from the $\mathbf{T}_{best}$ vector (i.e., $\mathbf{T}_{rank=1}$, which they have indicated matched the target geology) and the $\mathbf{T}_{target}$ vector (which by definition should match the target geology). The expert was then prompted to rank this new population as usual. If the expert ranked the realisations generated using $\mathbf{T}_{target}$ as better than those generated using $\mathbf{T}_{best}$, this indicates that the expert could potentially identify the realisations generated using $\mathbf{T}_{target}$ as more similar to $\mathbf{g}_{target}$ than those generated using $\mathbf{T}_{best}$. Thus we diagnose that they were not justified in pressing the 'Match' button as their best estimate of the statistics is still not a good enough match to the target image. If there was no preferential ranking, this indicates that the expert truly could not distinguish between the realisations generated by the $\mathbf{T}_{best}$ vector (found by them using the algorithm) and those generated using $\mathbf{T}_{target}$, in terms of their geology. Thus we diagnose that they were justified in pressing the 'Match' button.

We implemented this so-called *consistency test* only in the second stage of the algorithm. After the expert pressed 'Match' at this stage a population of realisations was presented to them where 3 out of the 6 realisations were generated with $\mathbf{T}_{target}$ and the remaining 3 of the 6 were generated using $\mathbf{T}_{best}$. The presentation of these realisations was precisely the same as with any other population at a 'normal' iteration in the algorithm. The test is based on the assumption that if the expert can distinguish between the realisations created using $\mathbf{T}_{target}$ and those created using $\mathbf{T}_{best}$, then they will rank the former set of realisations as $\{1, 2, 3\}$ and the latter as $\{4, 5, 6\}$. Conversely, if the expert could truly not distinguish between the realisations then any ranking would simply be due to random chance, and the probability of randomly ranking the population in this way is 0.05. Thus we say that the expert's decision (that $\mathbf{g}_{best}$ matches $\mathbf{g}_{target}$) is *confirmed* if the expert ranks the population in any other way than that described above. However, if the expert does rank the population in this manner we say that the decision is *unconfirmed*. Thus we can classify any $\mathbf{T}_{best}$ obtained with the algorithm as being confirmed or unconfirmed using the consistency test.

In order to collect more data on its performance, the algorithm was not terminated immediately in the second stage after the expert pressed the 'Match' button (and the consistency test was made). Instead, the algorithm was allowed to continue for a fixed number of iterations (20) in the second stage. Thus whenever the 'Match'

button was pressed the consistency test would be run but after its completion the algorithm would continue using the population found before the test was performed, during which period further matches could be identified and tested for consistency. This allowed us to build up an ensemble of $\mathbf{T}_{best}$ vectors, along with information as to whether the match had been confirmed or not.

In this demonstration of the algorithm we also sought to investigate the effect of the mutation parameters on the dynamics of the algorithm. Although in development we had found that values of $\beta_1 \sim 0.4$, $\alpha_1 \sim 0.4$, $\beta_2 \sim 0.2$ and $\alpha_2 \sim 0.2$, were sufficient to permit convergence, we varied these slightly between the experts tested; for each of the 12 experts these parameters were sampled from Uniform probability distributions on the following discrete sample spaces: $\beta_1 \in \{0.35, 0.4, 0.45\}$ and $\alpha_1 \in \{0.35, 0.45, 0.5\}$ and $\beta_2 \in \{0.1, 0.15, 0.2, 0.25\}$ and $\alpha_2 \in \{0.15, 0.2\}$.

5.7 Results

At each iteration of the algorithm the current population of statistics vectors $\mathcal{S}_l$, the current population of realisations $\mathcal{R}_l$, and the rankings provided by the experts were recorded. If the expert pressed 'Match' during the second stage $\mathbf{T}_{best}$ was recorded along with whether it was a confirmed or unconfirmed match using the consistency test. The root-mean-square error (RMSE) between the highest ranked statistics vector $\mathbf{T}_{rank=1}$ at each iteration and the target statistics vector $\mathbf{T}_{target}$ was also calculated at each iteration and recorded. The RMSE is defined as

$$\mathrm{RMSE}(\mathbf{T}_{rank=1}, \mathbf{T}_{target}) = \left(\frac{(\mathbf{T}_{rank=1} - \mathbf{T}_{target})^T (\mathbf{T}_{rank=1} - \mathbf{T}_{target})}{|\mathcal{C}|} \right)^{\frac{1}{2}} \tag{5.5}$$

where the average is taken over each of the $|\mathcal{C}|$ elements (probabilities) in the statistics vector, $\mathbf{T}$.

Figures 5.4 to 5.6 summarise the results for the 6 experts who were given a 'crack-pore' target image. Figures 5.7 to 5.9 summarise the results for the 6 experts who were given a 'round-pore' target image. The RMSE values at each iteration are plotted for each expert along with an indication of the iteration of transition between the first and second stages of the algorithm. The plot also indicates the iterations at which the expert obtained a $\mathbf{T}_{best}$ vector (i.e., where they pressed the 'Match' button) and whether this was confirmed or not by the consistency test. The figures

show the $\mathbf{g}_{target}$ provided to the expert during the first stage of the algorithm along with the final $\mathbf{g}_{rank=1}$ found by the expert using the first stage (i.e., that which was the best ranked member of the three vectors used to generate the initial population of the second stage). The figure also shows the $\mathbf{g}_{target}$ provided to the expert during the second stage of the algorithm along with the confirmed $\mathbf{g}_{best}$ found by the expert with the lowest RMSE. If no confirmed $\mathbf{g}_{best}$ vectors were found by the expert then the unconfirmed $\mathbf{g}_{best}$ vector with the lowest RMSE is displayed. If no $\mathbf{g}_{best}$ (either confirmed or unconfirmed) vectors were found then no image is displayed here.

Each figure also contains a legend with the mutation parameters (α_1, β_1, α_2 and β_2) applied for that run of the algorithm. As stated earlier we allowed these parameters to vary slightly for different experts. However, we found little meaningful correlation between these parameters and the minimum RMSE obtained by the experts (the absolute correlation coefficients between any of these parameters < 0.1). The legend also contains information about the expert's microscope experience; after discussion with the experts, each was given a score out of 10 indicating their microscope experience (with 0 indicating "no experience" and 10 indicating "very regular use"). Again, we found no significant correlation between this parameter and the minimum RMSE obtained by the experts (the absolute correlation coefficient between this microscope experience score and the minimum RMSE was < 0.1).

5.8 Discussion

At the end of the first stage all 12 experts found a realisation which they believed had statistically the same pore-space geology as the target image. In the second stage almost all of the experts found images that they believed had statistically the same pore-space geology as the target image, and most of these matches were confirmed using the consistency test. Experts 2 and 7 were able to find pore-space images with geology matching the target image in the second stage but these matches were not confirmed by the consistency test. Expert 8 was unable to find any realisations that he/she believed matched the target geology. Experts 2, 7 and 8 might have benefited from being able to continue using the second stage GUI beyond 20 iterations since convergence behaviour can be observed in RMSE values during this stage, which may have been prematurely terminated.

There were considerable numerical differences between the $\mathbf{T}_{target}$ and $\mathbf{T}_{best}$ vec-

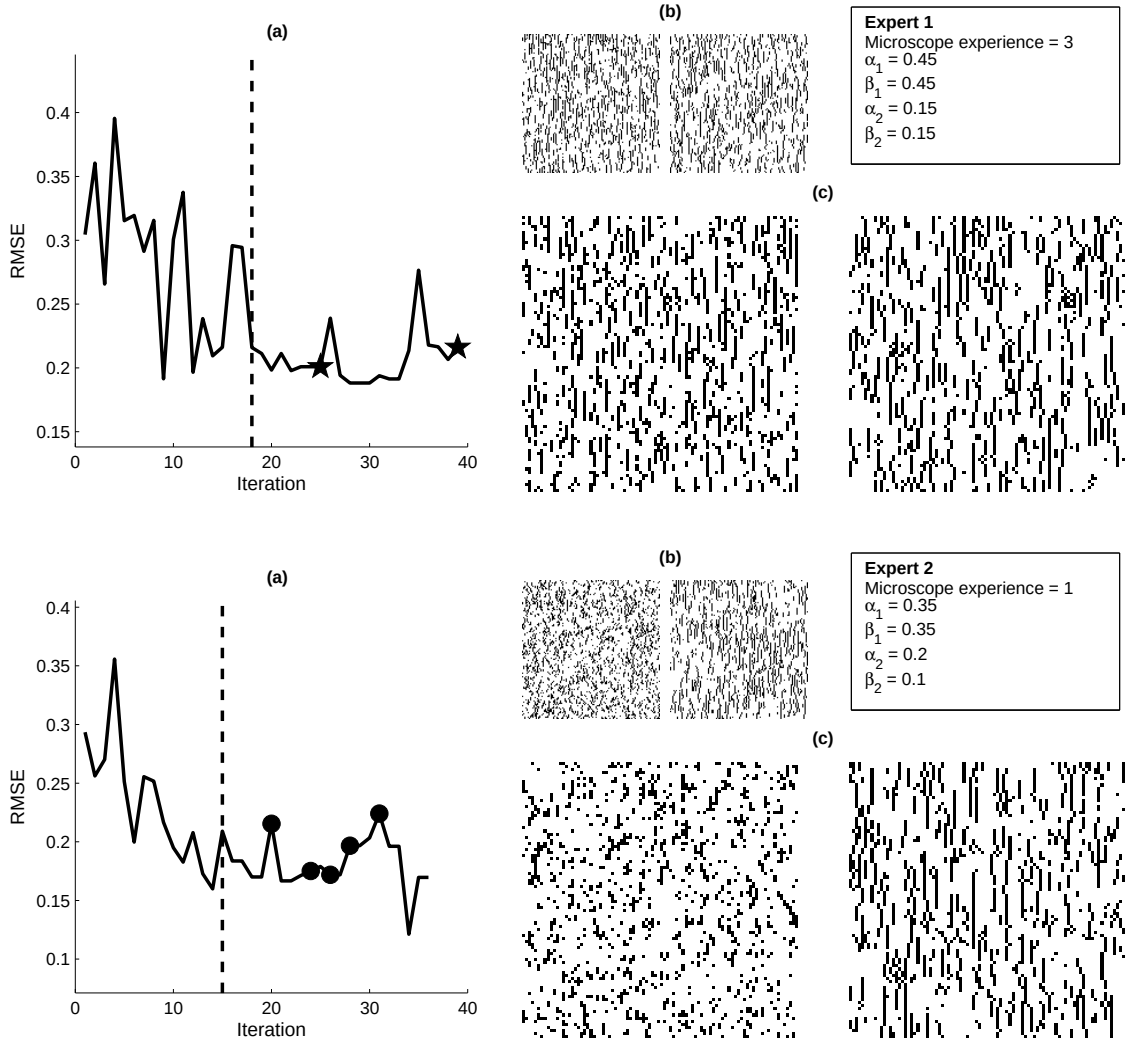

Figure 5.4: Results for experts 1 and 2. (a) The root-mean-square error (RMSE) between the highest ranked statistics vector $\mathbf{T}_{rank=1}$ at each iteration and the target statistics vector $\mathbf{T}_{target}$. The dashed line represents the transition from the first to second stage of the algorithm. The ● and ★ symbols represent an unconfirmed and confirmed $\mathbf{T}_{best}$ (or equivalently, match between $\mathbf{g}_{best}$ and $\mathbf{g}_{target}$), respectively. (b) (right) The target image $\mathbf{g}_{target}$ provided to the expert in the first stage, and (left) the $\mathbf{g}_{rank=1}$ realisation found at the end of the first stage. (c) (right) The target image $\mathbf{g}_{target}$ provided to the expert in the second stage, and (left) the confirmed $\mathbf{g}_{best}$ found by the expert with lowest RMSE in the second stage. For expert 2 no confirmed $\mathbf{g}_{best}$ was found so the unconfirmed $\mathbf{g}_{best}$ with lowest RMSE is shown.

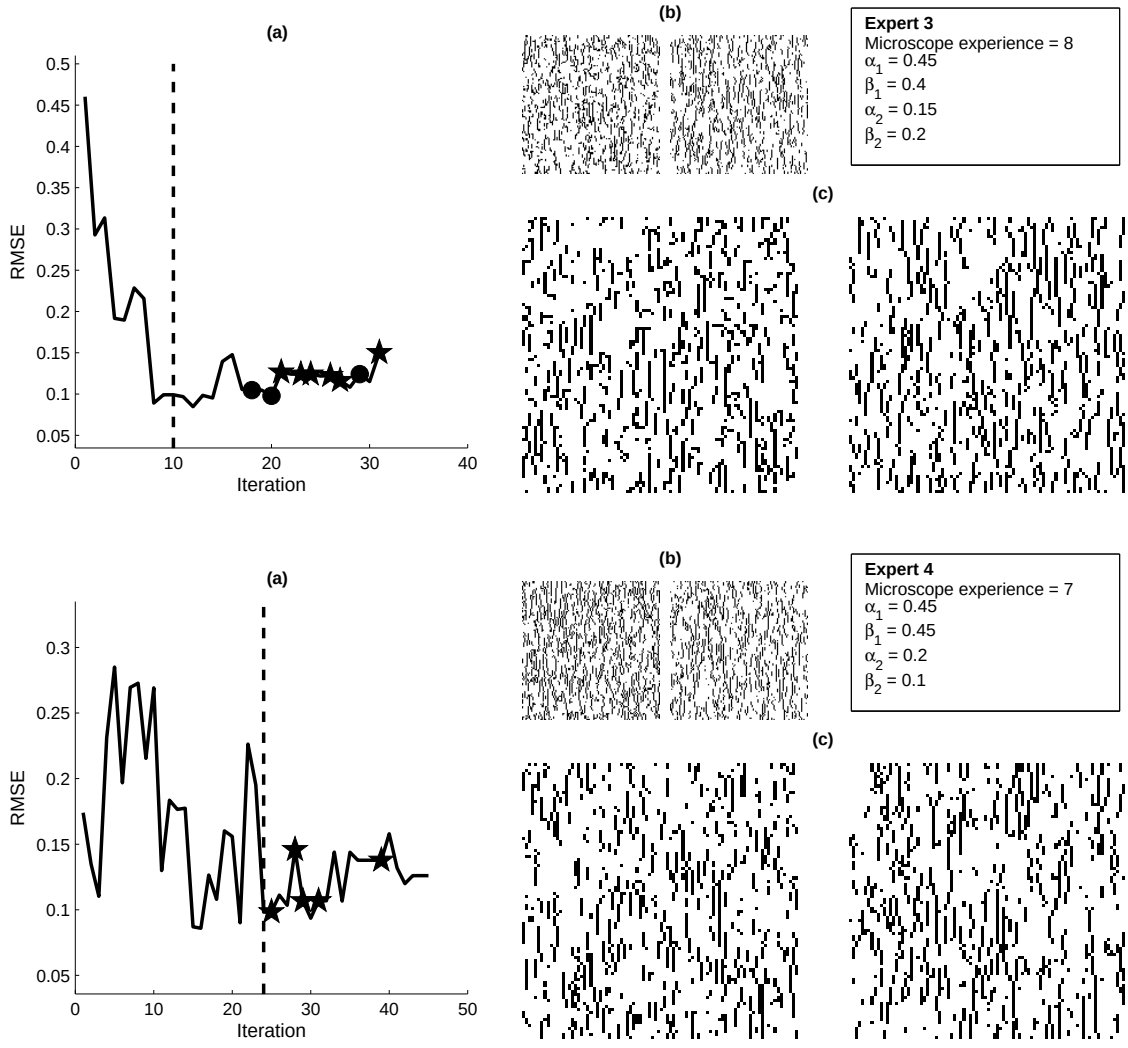

Figure 5.5: As for Figure 5.4, but for experts 3 and 4. Note that here both experts found confirmed $\mathbf{g}_{best}$ realisations in the second stage of the algorithm, so this is displayed on the left in (c).

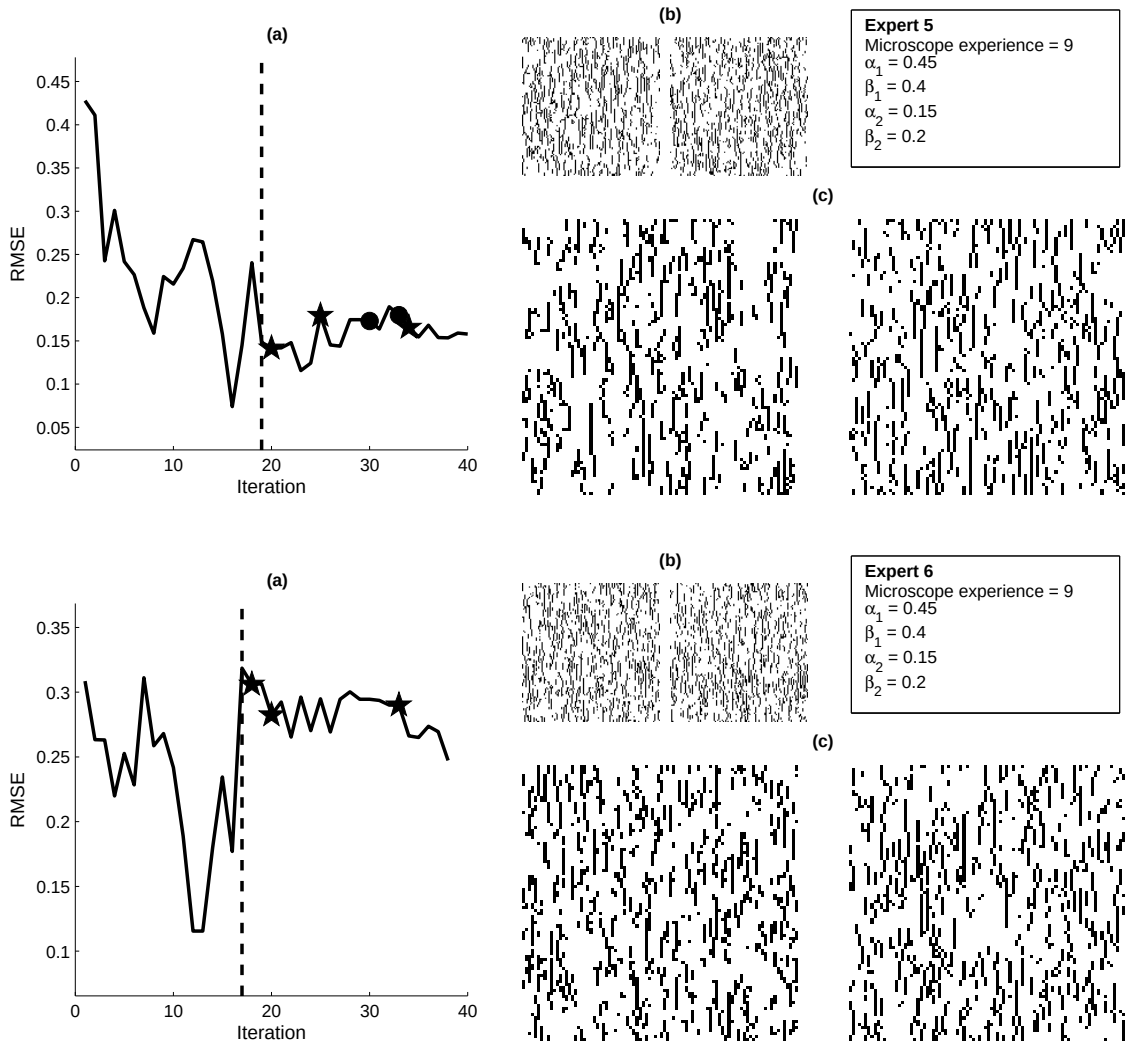

Figure 5.6: As for Figure 5.5, but for experts 5 and 6.

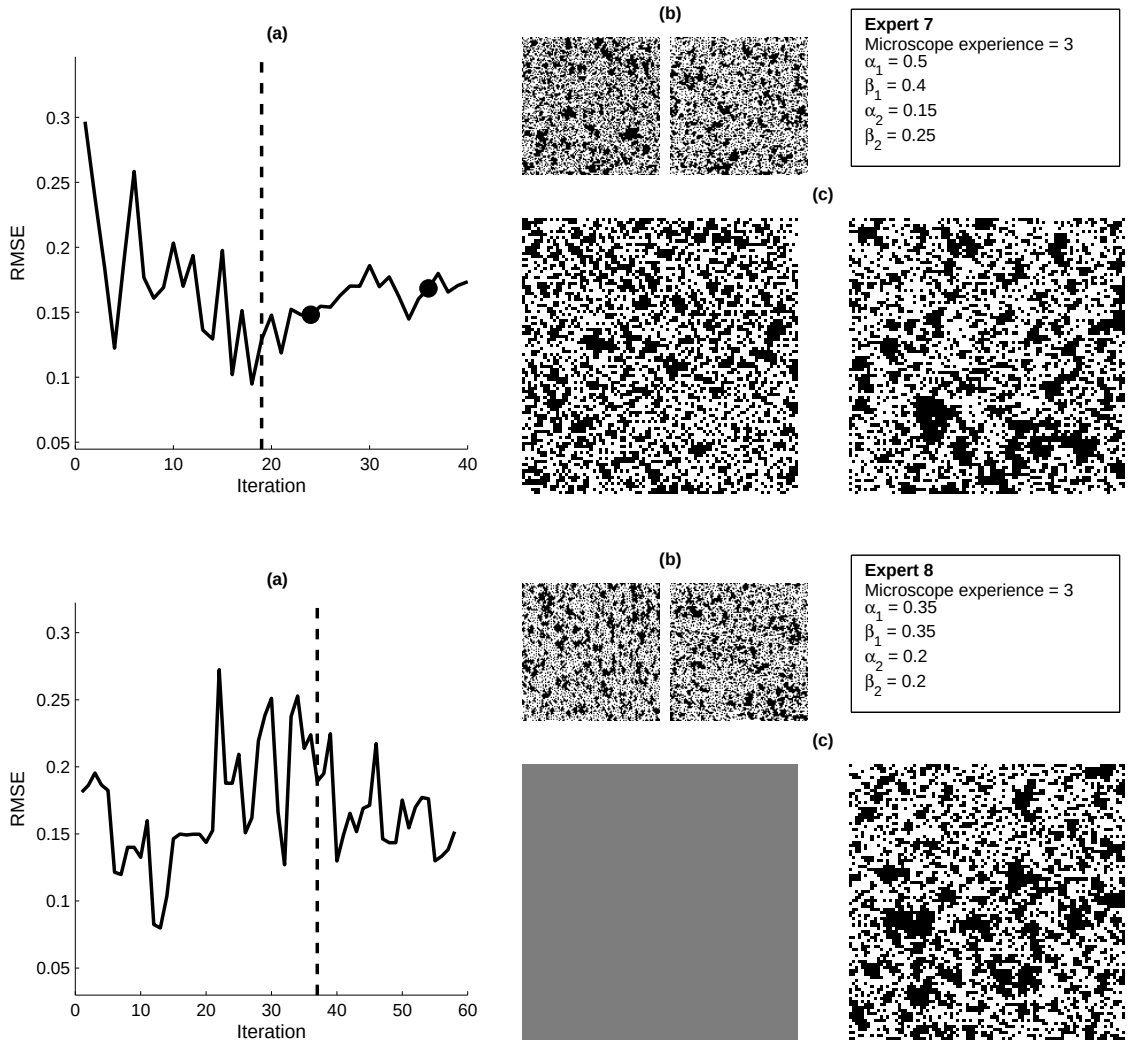

Figure 5.7: As for Figure 5.5, but for experts 7 and 8. Expert 7 found no confirmed $\mathbf{g}_{best}$ so the unconfirmed $\mathbf{g}_{best}$ with lowest RMSE is shown on the left in (c). Expert 8 found no $\mathbf{g}_{best}$ (i.e., neither unconfirmed or confirmed) so no $\mathbf{g}_{best}$ is shown for this expert in (c). Both of these experts would probably have benefited from being allowed to continue beyond 20 iterations in the second stage of the algorithm.

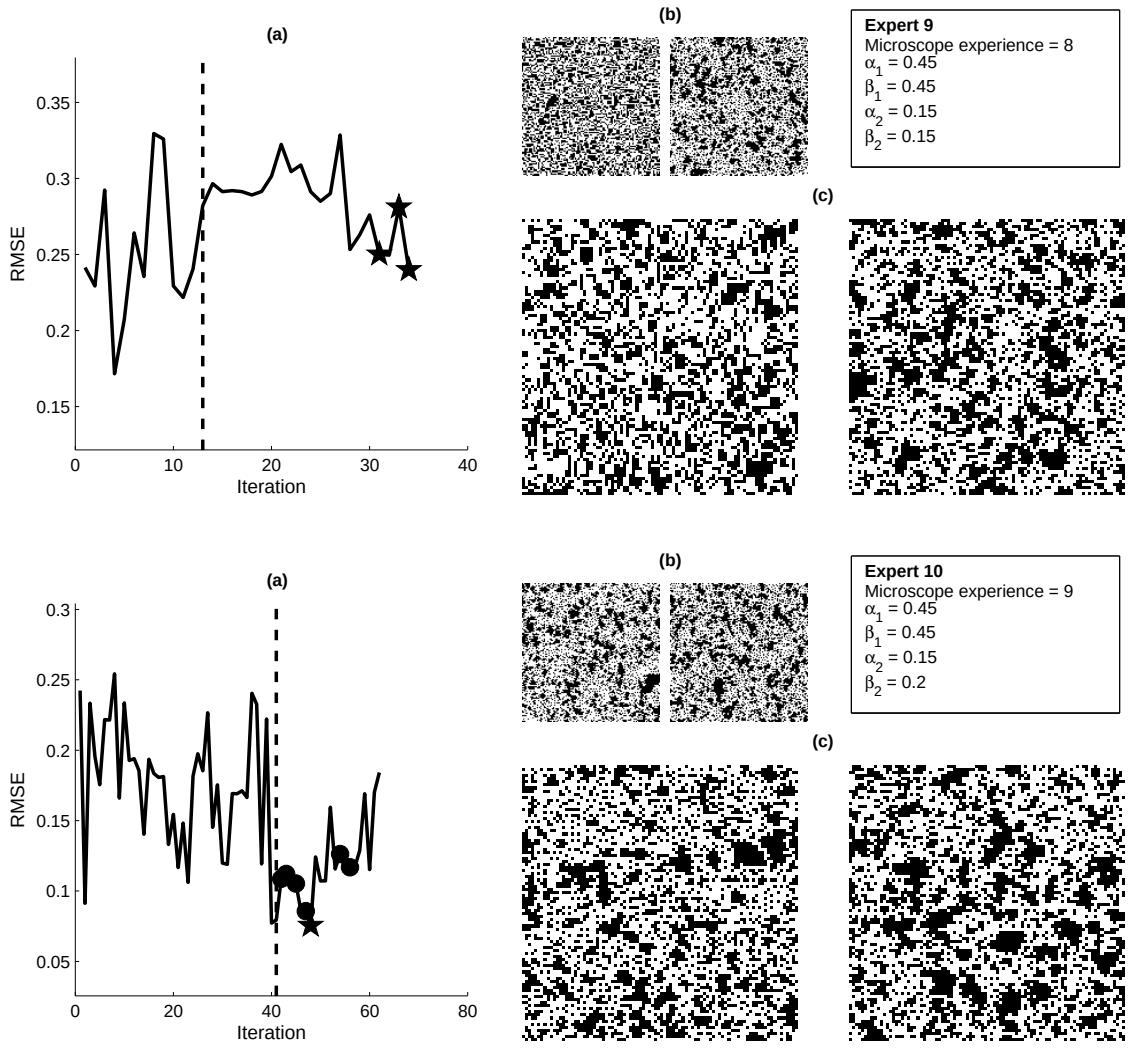

Figure 5.8: As for Figure 5.5, but for experts 9 and 10.

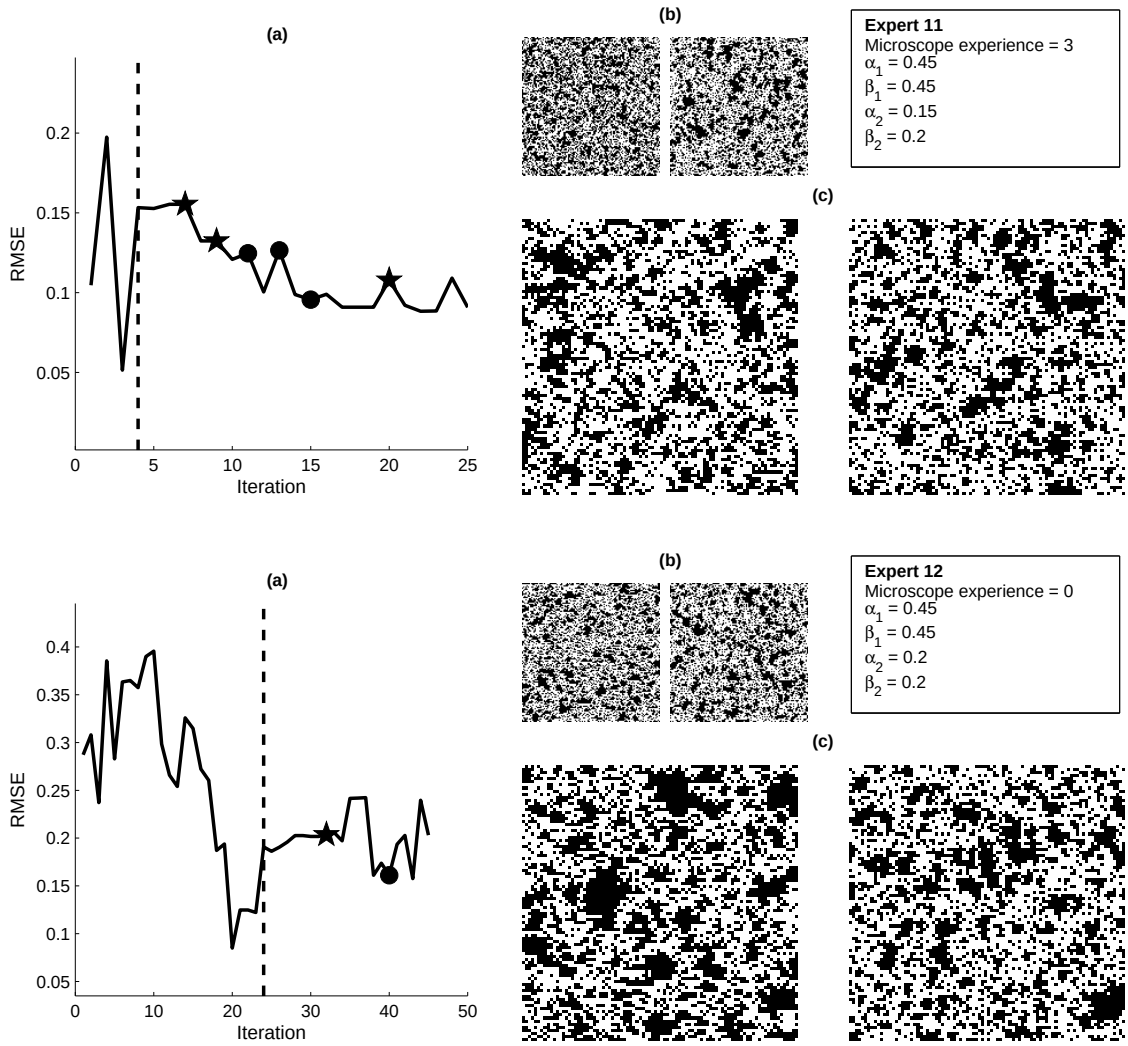

Figure 5.9: As for Figure 5.5, but for experts 11 and 12.

tors. The lowest RMSE values in these statistics (which are probabilities) being greater than 0.07 for all experts. Lower RMSE values (as low as 0.05) were obtained for some $\mathbf{T}_{rank=1}$ vectors but these, in the opinion of the expert, did not produce realisations which matched the target geology (the expert did not press the 'Match' button after ranking these realisations, thus they are not classed as ideal statistic estimates $\mathbf{T}_{best}$). In any case, given that these are probabilities, this is a significant error. It demonstrates that there may be a limit in the ability of experts to discriminate between images with different spatial statistics using this algorithm. In an inverse problem this feature of the solution would be called the null-space (Gubbins, 2004, p. 110). It is important to have identified such a 'geological null-space' because although the expert may not be able to discriminate between different spatial statistics (or in practice, the realisations created using those statistics), these differences may be of importance to the application for which our geostatistical model has been developed. For example, in the context of seismic inversion $\mathbf{g}$ might be used to model the distribution of facies in a reservoir, and small differences in their spatial distribution may be in-discriminable by experts but may cause large differences in the flow characteristics of the reservoir model as a whole (Tsang, 1984).

In principle there is another possible explanation for the large RMSE values which is totally unrelated to the experts' abilities. It is possible that, for a certain $\mathbf{T}$ vector, the produced realisations will virtually never contain a certain spatial configuration ($C \in \mathcal{C}$, say) of the pore-space variable within the neighbourhood of cells, if the probability of that certain configuration occurring is extremely low. This would imply that the corresponding probability $p(g_i|\mathbf{g}_{Ne(i)} = C)$, that is the corresponding t_k in $\mathbf{T}_{target}$, cannot have any effect on the rock pore-space image realisation produced using that $\mathbf{T}_{target}$ vector. Thus the expert may find a $\mathbf{T}_{best}$ vector which produces images which almost perfectly match the target, but which have a completely different value for this statistic. However, in our tests we ensured that this was not the case when designing the particular $\mathbf{T}_{target}$ vectors (both the 'crack-pore' and 'round-pore' varieties) that we used: we checked that all possible neighbourhood configurations (that is all elements in $\mathcal{C}$) occurred frequently within any realisation of $\mathbf{g}$ produced using the target statistics vector $\mathbf{T}_{target}$. Hence it is very unlikely that this is the cause of the final residual misfit which we observe.

Similarly, the limit is almost certainly not controlled by the mutation parameters. Whilst it might be expected that larger values for α and β would cause the minimum RMSE values to be large, since these parameters are interpreted as controlling the

'step' length of the algorithm, we do not see any significant positive correlation between the RMSE measures and these parameters. The fact that many of the expert's matches were confirmed by the consistency test indicates that the cause is more likely to be intrinsic to the expert. This is certainly not to say that the limit is *equal* to the intrinsic limit of the expert, but it is likely to be related to it.

In any case, it appears that experts are only able to discriminate between the probabilities used in this geostatistical model at a minimum level of ~ 0.1, or 10%, using the algorithm developed here. This typical level of error is illustrated by Figure 5.10 which shows a histogram of the minimum RMSE values obtained by the experts. This would imply that there is a significant null-space in the experts' abilities to choose between statistics (and by implication, between different statistical models). These results may be able to be improved if formal rules of statistical expert elicitation theory (e.g., Choy et al., 2009; Knol et al., 2010; Truong et al., 2013) are applied. Such rules (procedures) aim to provide a framework for elicitation experiments such that bias in estimates of expert knowledge about a variable (in our case the $\mathbf{T}$ probabilities) is minimised. Despite the measures which we have taken to try to ensure that the expert finds $\mathbf{g}_{best}$ realisations which truly match the target geology (e.g., by using the consistency test) we have not considered explicitly the bias which the user may have prior to, or develop during, the algorithm with respect to how they compare different realisations. It is this type of bias which elicitation theory aims to remove. Furthermore, formal elicitation theory could be used to combine the opinion of multiple experts to obtain one single estimate of the ideal statistics (Baddeley et al., 2004; Polson and Curtis, 2010; Allard et al., 2012). Experts could be asked to rank each $\mathbf{g}_j$ in a population as a group or individually, and their resulting ranks combined. Other forms of information might also be elicited from the experts rather than just visual comparisons, such as numerical information. Additionally, further constraints may be derived from physical measurements or knowledge (and hence modelling) of geological processes.

The two-stage GUI implementation has produced interesting results in itself. In Figures 5.4 to 5.9 it can be observed that it is actually quite rare for the RMSE measure to be reduced significantly in the second 'fine tuning' stage, compared to the RMSE in the first stage (which was designed with only the intention of obtaining a good starting population). This may be due to the expert concentrating on different aspects of the geology depending upon the magnification of realisations presented to them. At increased magnification the expert can pick out some fine details (such

as complex shapes) more easily, but at lower magnifications the human eye may more efficiently judge bulk statistical properties (such as the overall pore-matrix proportions). Thus after the transition to the greater magnification, the loss of the expert's ability to evaluate these bulk features may in fact result in an increase in RMSE. Nevertheless, we found this transition to be a valuable component of our elicitation procedure as the rapid rate of convergence in the first stage reduced expert fatigue, and thus also its concomitant biases. In future, it may be interesting to trial a GUI which displays both low and high magnifications *simultaneously* to the expert.

We have shown that the direct elicitation of spatial statistics from a geological expert is possible using the elicitation method. These spatial statistics may be used to specify a geostatistical model which defines $p(\mathbf{g})$. It is an important feature of the algorithm employed that it allows the expert to interact directly with the optimisation without having to understand the underlying details of the geostatistical model. In our example, the expert does not have to deal explicitly with probabilities, and is instead able to concentrate on their area of expertise - the analysis of (spatial) geological features. The example model which we have used is a practically employed multi-point geostatistical model used in both petroleum (Kjønsberg and Kolbjørnsen, 2008; Okabe and Blunt, 2004; van der Land et al., 2013) and soil geostatistics (Wu et al., 2004; Zhu et al., 2007; Li, 2007), and our algorithm has immediate practical relevance for determining parameters for such applications. The particular implementation of this model is quite parsimonious, however; the number of free parameters in the model is quite low ($|\mathcal{C}| = 8$) compared to other multi-point geostatistical models.

There is an inherent advantage in having a smaller number of model parameters since it means that exploring the parameter space and hence finding the ideal statistics is easier using the genetic algorithm. Thus we avoid the so-called 'curse of dimensionality' which effects many optimisation methods in higher dimensional model spaces (Curtis and Lomax, 2001): the volume of the space to be explored grows exponentially with the number of free parameters to be determined. As the number of dimensions increases, as much as exponentially many more iterations might be required to find the ideal statistics. This would be particularly problematic in this case since the algorithm requires human input upon each iteration. Thus it is likely that the elicitation method will be significantly more costly (in terms of expert time) when applied to geostatistical models which require a large number of statistics (such as full conditional distributions with large neighbourhood structures).

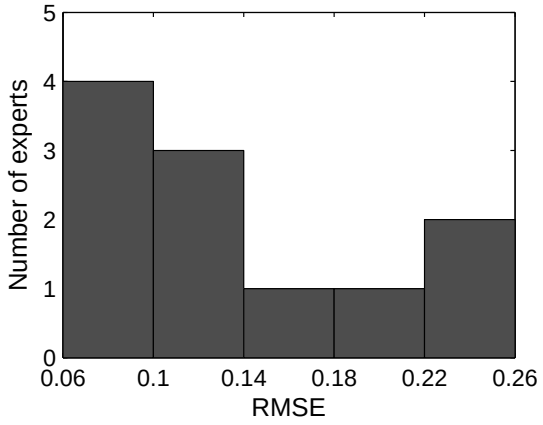

Figure 5.10: A histogram of the lowest root-mean-square error (RMSE) between $\mathbf{T}_{best}$ and the target statistics vector $\mathbf{T}_{target}$, found by each expert. For each expert the confirmed $\mathbf{T}_{best}$ match with the lowest RMSE was used. For experts 2 and 7 no confirmed $\mathbf{T}_{best}$ match was found, thus the unconfirmed $\mathbf{T}_{best}$ match with the lowest RMSE was used. For expert 8 neither an unconfirmed nor confirmed $\mathbf{T}_{best}$ match was found, thus no RMSE value for that expert is included in this histogram.

Although we demonstrated the elicitation method for a geostatistical model of a rock's pore-space, it can be immediately applied to determine statistics which can be used to specify the geological prior distribution $p(\mathbf{g})$ used in seismic inversion. We showed in section 5.6.1 that the pore-space model is a particular instance of the multi-point geostatistical model as described using full conditionals, thus it may itself be applied in seismic inversion. However, the elicitation algorithm remains untested for eliciting full conditional probabilities in their *general* form (equation 4.1). In this case efficient sequential sampling $\mathbf{g} \sim p(\mathbf{g}|\mathbf{T})$ would not be possible, thus the elicitation algorithm may be slowed significantly. However, the algorithm may be immediately applied to find the parameters of two-point statistical models such as variograms since these typically require few defining statistics and can be sampled from very rapidly (Caers, 2005, pp. 21-29).

5.9 Summary

We have shown that spatial statistics can be elicited directly from a geological expert using an elicitation methodology based on the use of genetic algorithms. The algorithm iteratively updates a population of candidate statistics vectors, using an expert's opinion of how well realisations generated with those statistics (using the geostatistical model) match their envisagement of the appropriate spatial relationships between the geological features. Thus, the algorithm allows experts to interact directly with the statistical optimisation without having to understand the details of the underlying geostatistical model.

The algorithm was used to estimate the statistics of a multi-point geostatistical model, parametrised using conditional probabilities. 12 experts were asked to use the algorithm to find the statistics suitable for representing a target pore-space image. The image had known statistics, thus numerical convergence towards the true answer could be calculated and monitored. 11 of the 12 experts were able to obtain a match they deemed reasonable. Convergence rates were acceptable, with most experts taking less than 40 iterations to find a matching realisation. This experiment also assesses the intrinsic human uncertainty in comparing spatial statistics when using the algorithm described. We found that there was a large misfit between the ideal statistics (found by the expert) and the known statistics (those used to generate the target image). The minimum root-mean-square error was typically $>$ 0.1 for most experts. These errors are large considering the statistics were defined as probabilities. More accurate discrimination is therefore likely to require information obtained from complementary elicitation techniques, physical measurements or knowledge of processes.

The method developed is general and may be immediately extended to the estimation of the parameters of other geostatistical models such as variograms. In theory it is also possible to use the method to estimate the probabilities in full conditional distributions. Thus this new elicitation method can potentially be used to determine the geological prior distribution $p(\mathbf{g})$ used in seismic inversion.

References

Allard, D., A. Comunian, and P. Renard (2012), Probability aggregation methods in geoscience, *Mathematical Geosciences*, *44*(5), 545–581.

Baddeley, M. C., A. Curtis, and R. Wood (2004), An introduction to prior information derived from probabilistic judgements: elicitation of knowledge, cognitive bias and herding, *Geological Society, London, Special Publications*, *239*(1), 15–27.

Bond, C., A. Gibbs, Z. Shipton, and S. Jones (2007), What do you think this is? "Conceptual uncertainty" in geoscience interpretation, *GSA today*, *17*(11), 4.

Bond, C., R. Lunn, Z. Shipton, and A. Lunn (2012), What makes an expert effective at interpreting seismic images?, *Geology*, *40*(1), 75–78.

Boschetti, F., and L. Moresi (2000), Comparison between interactive (subjective) and traditional (numerical) inversion by genetic algorithms, in *Proceedings of the 2000 Congress on Evolutionary Computation, 2000.,* pp.522–528, IEEE.

Boschetti, F., and L. Moresi (2001), Interactive inversion in geosciences, *Geophysics*, *66*(4), 1226–1234.

Caers, J. (2005), *Petroleum geostatistics*, Richardson, TX: Society of Petroleum Engineers.

Caers, J., S. Srinivasan, and A. Journel (1999), Geostatistical quantification of geological information for a fluvial-type North Sea reservoir, in *SPE Annual Technical Conference and Exhibition*.

Choy, S. L., R. O'Leary, and K. Mengersen (2009), Elicitation by design in ecology: using expert opinion to inform priors for Bayesian statistical models, *Ecology*, *90*(1), 265–277.

Comunian, A., P. Renard, J. Straubhaar, and P. Bayer (2011), Three-dimensional high resolution fluvio-glacial aquifer analog. Part 2: Geostatistical modeling, *Journal of Hydrology*, *405*(1), 10–23.

Cui, H., A. Stein, and D. E. Myers (1995), Extension of spatial information, Bayesian kriging and updating of prior variogram parameters, *Environmetrics*, *6*(4), 373–384.

Curtis, A. (2012), The science of subjectivity, *Geology*, *40*(1), 95–96.

Curtis, A., and A. Lomax (2001), Prior information, sampling distributions, and the curse of dimensionality, *Geophysics*, *66*(2), 372–378.

Curtis, A., and R. Wood (2004), Geological Society of London.

David, M., and R. Blais (1977), Geostatistical ore reserve estimation, *Developments in geomathematics*.

Dimitrakopoulos, R. (1998), Conditional simulation algorithms for modelling orebody uncertainty in open pit optimisation, *International Journal of Surface Mining, Reclamation and Environment*, *12*(4), 173–179.

Dueholm, K., and T. Olsen (1993), Reservoir analog studies using multimodel photogrammetry: a new tool for the petroleum industry, *AAPG Bulletin*, *77*(12), 2023–2031.

Goldberg, D. E. (1989), Genetic algorithms in search, optimization, and machine learning.

Gubbins, D. (2004), *Time series analysis and inverse theory for geophysicists*, Cambridge University Press.

Hill, J., D. Tetzlaff, A. Curtis, and R. Wood (2009), Modeling shallow marine carbonate depositional systems, *Computers & Geosciences*, *35*(9), 1862–1874.

Honarkhah, M., and J. Caers (2010), Stochastic simulation of patterns using distance-based pattern modeling, *Mathematical Geosciences*, *42*(5), 487–517.

James, A., S. L. Choy, and K. Mengersen (2010), Elicitator: An expert elicitation tool for regression in ecology, *Environmental Modelling & Software*, *25*(1), 129–145.

Journel, A., R. Gundeso, E. Gringarten, and T. Yao (1998), Stochastic modelling of a fluvial reservoir: a comparative review of algorithms, *Journal of Petroleum Science and Engineering*, *21*(1), 95–121.

Keehm, Y., T. Mukerji, and A. Nur (2004), Permeability prediction from thin sections: 3D reconstruction and Lattice-Boltzmann flow simulation, *Geophysical Research Letters*, *31*(4).

Kerry, R., and M. Oliver (2007), Comparing sampling needs for variograms of soil properties computed by the method of moments and residual maximum likelihood, *Geoderma*, *140*(4), 383–396.

Kjønsberg, H., and O. Kolbjørnsen (2008), Markov mesh simulations with data conditioning through indicator kriging, in *Proceedings of the 8th International Geostatistics Congress, Santiago, Chile*.

Knol, A. B., P. Slottje, J. P. van der Sluijs, and E. Lebret (2010), The use of expert elicitation in environmental health impact assessment: a seven step procedure, *Environmental Health*, *9*(1), 19.

Kupfersberger, H., and C. Deutsch (1999), Methodology for integrating analog geologic data in 3-D variogram modeling, *AAPG bulletin*, *83*, 1262–1278.

Kynn, M. (2008), The heuristics and biases in expert elicitation, *Journal of the Royal Statistical Society: Series A Statistics in Society*, *171*(1), 239–264.

Leuangthong, O., J. A. McLennan, and C. V. Deutsch (2004), Minimum acceptance criteria for geostatistical realizations, *Natural Resources Research*, *13*(3), 131–141.

Li, W. (2007), Transiograms for characterizing spatial variability of soil classes, *Soil Science Society of America Journal*, *71*(3), 881–893.

Lindley, D. (1983), Reconciliation of probability distributions, *Operations Research*, *31*(5), 866–880.

Lindley, D. V., A. Tversky, and R. V. Brown (1979), On the reconciliation of probability assessments, *Journal of the Royal Statistical Society. Series A (General)*pp.146–180.

Loquin, K., and D. Dubois (2010), Kriging and epistemic uncertainty: a critical discussion, in *Methods for Handling Imperfect Spatial Information,* pp.269–305, Springer.

Matheron, G. (1963), Principles of geostatistics, *Economic geology*, *58*(8), 1246–1266.

Michael, H., H. Li, A. Boucher, T. Sun, J. Caers, and S. Gorelick (2010), Combining geologic-process models and geostatistics for conditional simulation of 3-D subsurface heterogeneity, *Water Resources Research*, *46*(5).

Nordahl, K., P. S. Ringrose, and R. Wen (2005), Petrophysical characterization of a heterolithic tidal reservoir interval using a process-based modelling tool, *Petroleum Geoscience*, *11*(1), 17–28.

Okabe, H., and M. J. Blunt (2004), Prediction of permeability for porous media reconstructed using multiple-point statistics, *Physical Review E*, *70*(6), 066,135.

Okabe, H., and M. J. Blunt (2005), Pore space reconstruction using multiple-point statistics, *Journal of Petroleum Science and Engineering*, *46*(1), 121–137.

Polson, D., and A. Curtis (2010), Dynamics of uncertainty in geological interpretation, *Journal of the Geological Society*, *167*(1), 5–10.

Price, D., A. Curtis, and R. Wood (2008), Statistical correlation between geophysical logs and extracted core, *Geophysics*, *73*(3), E97–E106.

Pringle, J., A. Westerman, J. Clark, N. Drinkwater, and A. Gardiner (2004), 3D high-resolution digital models of outcrop analogue study sites to constrain reservoir model uncertainty: an example from Alport Castles, Derbyshire, UK, *Petroleum Geoscience*, *10*(4), 343–352.

Pringle, J., J. Howell, D. Hodgetts, A. Westerman, and D. Hodgson (2006), Virtual outcrop models of petroleum reservoir analogues: a review of the current state-of-the-art, *First Break,* p.33.

Ringrose, P., G. Pickup, J. Jensen, and M. Forrester (1999), The Ardross reservoir gridblock analog: sedimentology, statistical representivity, and flow upscaling, in *AAPG Memoir 71: Reservoir Characterization-Recent Advances*, edited by R. A. Schatzinger and J. F. Jordan, pp.265–276, AAPG.

Stien, M., and O. Kolbjørnsen (2011), Facies modeling using a Markov mesh model specification, *Mathematical Geosciences*, *43*(6), 611–624.

Truong, P. N., and G. Heuvelink (2013), Uncertainty quantification of soil property maps with statistical expert elicitation, *Geoderma*, *202*, 142–152.

Truong, P. N., G. Heuvelink, and J. P. Gosling (2013), Web-based tool for expert elicitation of the variogram, *Computers & Geosciences*, *51*, 390–399.

Tsang, Y. (1984), The effect of tortuosity on fluid flow through a single fracture, *Water Resources Research*, *20*(9), 1209–1215.

Tversky, A., and D. Kahneman (1974), Judgment under uncertainty Heuristics and biases, *science*, *185*(4157), 1124–1131.

van der Land, C., R. Wood, K. Wu, M. I. van Dijke, Z. Jiang, P. W. Corbett, and G. Couples (2013), Modelling the permeability evolution of carbonate rocks, *Marine and Petroleum Geology*, *48*, 1–7.

Wood, R., and A. Curtis (2004), Geological prior information and its applications to geoscientific problems, *Geological Society, London, Special Publications*, *239*(1), 1–14.

Wu, K., N. Nunan, J. Crawford, I. Young, and K. Ritz (2004), An efficient Markov chain model for the simulation of heterogeneous soil structure, *Soil Science Society of America Journal*, *68*(2), 346–351.

Wu, K., M. Van-Dijke, G. Couples, Z. Jiang, J. Ma, K. Sorbie, J. Crawford, I. Young, and X. Zhang (2006), 3D stochastic modelling of heterogeneous porous media–applications to reservoir rocks, *Transport in porous media*, *65*(3), 443–467.

Zhang, T., D. Lu, and D. Li (2009), Porous media reconstruction using a cross-section image and multiple-point geostatistics, in *International Conference on Advanced Computer Control, 2009. ICACC'09.,* pp.24–29, IEEE.

Zhu, L., B. Ma, L. Zhang, and L. Zhang (2007), The study of distribution and fate of nitrobenzene in a water/sediment microcosm, *Chemosphere*, *69*(10), 1579–1585.

Chapter 6

Discussion

6.1 Overview

Each of Chapters 2-5 has discussed, and offered a solution to, one of the research questions posed in section 1.8. The purpose of this chapter is to discuss the overall implications of the methodologies developed, and to identify topics for future work. In section 6.2 we discuss how successfully each of the individual methodologies can be incorporated into the two-stage Bayesian seismic inversion workflow described in section 1.6. In section 6.3 we then discuss whether any of the methodologies developed here have the potential to go beyond this two-stage inversion approach, and permit an efficient 'single-stage' Bayesian seismic inversion method.

6.2 Integration of methods into the two-stage inversion approach

In theory, all of the methods described in this thesis have a defined role within the two-stage inversion workflow, as illustrated in Figure 1.1. The deep neural network methodology (Chapter 2) may be used to improve the fidelity of the prior information which is included in the elastic inversion portion of the method. Prior replacement (Chapter 3) may be used to vary the prior which is implicit within cell-wise geological inversion solutions (i.e., posterior estimates) in order to permit a spatially varying prior. The results of neural network inversion can then be used within a stochastic geological inversion methodology, such as the recursive algorithm or Gibbs sampling

methodologies. The recursive algorithm (Chapter 4) permits exact sampling from the geological posterior, with a prior defined using multi-point geostatistics, without the need for potentially biased MCMC sampling. Finally, the elicitation algorithm (Chapter 5) can potentially be used to determine the appropriate statistics for the multi-point geostatistical model used in the chosen stochastic geological inversion method (such as the recursive algorithm), directly from a geological expert without the need for the production of a training image. However, there are numerous limitations to these new methodologies which may restrict their immediate applicability within the two-stage inversion method.

The definition of the deep neural network methodology in Chapter 2 is quite restricted. The neural network was effectively defined as a 1-D recursive filter to be applied, in isolation, down single traces of elastic parameter estimates obtained from deterministic elastic inversion. Thus 2-D or 3-D lateral correlations are not accounted for in its predictions. However, adding additional prior information about the lateral correlations is not strictly necessary; in the example application to the Laggan dataset lateral continuity exists in the elastic parameter estimates after transformation using the deep neural network operator, despite the fact that this operator is 1-D in nature. This is because information about lateral correlations is introduced in the original low-fidelity prior employed in deterministic elastic inversion. Thus extending the neural network to function recursively in 2 or 3 dimensions is not immediately necessary for this methodology to be applied to practical problems. However, such an extension is necessary if we wish to apply high-fidelity prior information about the lateral correlations (rather than just about the vertical ones).

What is more, we have only demonstrated the deep neural network method for a relatively simple 1-D model (comprising few layers, for the Laggan dataset application). We have not proven its worth for more complex 1-D models (i.e., containing more layers with more complex thickness relations, perhaps related to sequence stratigraphic concepts). However, the results did demonstrate that the neural network learnt the general 'concept' of layering; application of the neural network predicted three sand layers, whilst the model used to produce the training dataset had only two sand layers. This is encouraging since it suggests that more complex models (including 2- and 3-D models) may be learnt by taking advantage of the spatial repetition of similar geological features. Nevertheless more testing must be done to prove the method's worth for more complex models, and hence its use in practical two-stage seismic inversion. Because the deep neural network methodology was

developed in the latter stages of the project its results were not tested as input to the recursive algorithm method developed in Chapter 4. It would be interesting to test whether using the improved elastic inversion results within geological inversion would lead to an improvement in the latter inversion's results.

The prior replacement operation developed in Chapter 3 permits the results of efficient neural network inversion to be implemented within stochastic geological inversion (as shown in Chapter 4). However, the most interesting questions about prior replacement are concerned with its effect on the quality of the final estimate of the so-called new posterior distribution, given the number of samples used to determine the old posterior and the relative properties of the new and old prior distributions (see Appendix F). This question is perhaps of less interest in the context of practical Bayesian seismic inversion since in general the geological sample spaces (i.e., $\mathcal{G}$) within the individual cells are small, thus a very dense sampling over these small sample spaces can be afforded, and the quality of the posterior solution can be ensured regardless. However, as we have shown in Appendix F, prior replacement may be used as a variance reduction technique similar to importance sampling, in the context of general Bayesian inversion (where it is often the case that only a limited number of samples from the posterior may be available).

The recursive algorithm developed in Chapter 4 was shown to be a useful bias-free alternative to other Monte-Carlo techniques for determining the geological posterior. However, in practice it is limited by the local prior property: even the computational cost of the approximate version of the algorithm scales exponentially with the size of the neighbourhood of the full conditional distribution. We only demonstrated the algorithm for a 2-D subsurface model grid since this limitation is even more acute for 3-D grids. In section 4.9 we suggested that the algorithm could be designed to perform calculations on a geological sample space of reduced size ($\mathcal{G}'$), thus reducing the computational cost of the algorithm. However, there is no clear mathematical approach for developing such an approximation within the framework of the recursive algorithm. A practical approach should be developed to permit efficient application of the recursive algorithm with larger neighbourhood structures (as are commonly encountered in practical geological inversion problems).

In our example application of the recursive algorithm we ignored the conditions required by the Hammersley-Clifford theorem on the full conditional distributions: we simply assumed that the full conditionals obtained (using the 'event-counting' method) from the training image were correct, and compensated for the resulting

(small) errors in the calculated conditional probabilities by re-normalisation. We currently have little understanding of the effect of this approximation. Furthermore, we did not extend the algorithm to inversion for continuous geological parameters $\mathbf{m}_i$ (which are commonly the target in geological inversion). In principle this is not a difficult task: one need only replace g_i with $\mathbf{m}_i$, and summations with integrals, within the equations throughout Chapter 4. However, it is likely that the computational cost of the algorithm in the continuous case would be more difficult to control, and would require some method of parametrising the continuous geological parameter space.

In Chapter 5 it was shown that the direct expert elicitation method could potentially be used to obtain the $p(\mathbf{g})$ distribution used in geological inversion. However, there are two practical issues which prevent its immediate use in the two-stage workflow. The first of these is that it was only demonstrated practically for a particular geostatistical model (i.e., that in section 5.6.1). This model was a multi-point geostatistical model, as defined in section 1.5.2, but defined with non-symmetrical neighbourhood such that exact sequential sampling could be performed (to obtain realisations which could be presented rapidly to the expert). There is no theoretical reason why the method cannot be used to determine the probabilities in a full conditional distribution with symmetrical neighbourhood (i.e., equation 4.1). However, in this case sequential sampling could not be performed, and a Monte-Carlo sampling algorithm would have to be used to obtain realisations. This would slow the implementation of the elicitation method considerably, perhaps to the point where it would be impractical for the expert to interact with. Thus application to such multi-point geostatistical models (as used by the recursive algorithm in Chapter 4, for example) may not be immediately possible; further research is required to make the algorithm practical for general multi-point geostatistical models.

The second practical problem with the elicitation method is that the amount of time required by the algorithm for elicitation increases rapidly with the number of parameters (statistics) in the geostatistical model (and hence it was only demonstrated to work for a geostatistical model with a relatively low number of parameters). In the case of multi-point geostatistics this implies that the method may be restricted to full conditionals with small neighbourhood structures (either symmetric or non-symmetrical). Again, further work is required before this algorithm may be applied to the type of realistically-sized geostatistical models which are regularly used in geological inversion (i.e., full conditionals with large neighbourhoods).

6.3 Potential for new 'single-stage' method of inversion

Both the deep neural network (Chapter 2) and recursive algorithm (Chapter 4) methods offer novel approaches to the Bayesian seismic inversion problem. The deep neural network method completely avoids traditional Bayesian inversion techniques for elastic inversion since it is not based on the usual stochastic (i.e., MCMC) or deterministic (gradient-ascent) methods. The recursive algorithm offers a different Monte-Carlo technique, which avoids some of the bias issues associated with MCMC approaches. However, the recursive algorithm can only be applied to the geological inversion part of the two-stage inversion method, where the local likelihood property can be assumed. Thus it would not seem to offer any alternative to the overall two-stage seismic inversion approach which has been assumed in this thesis.

However, there is no such limitation on the deep neural network method; its predictions are based on data which is distributed across (that is, down a trace in) the model grid. As argued in Chapter 2, we could in principle train a deep neural network to emulate the mapping from the AVA-type data to the posterior over the elastic parameters, i.e., $\mathbf{d} \rightarrow p(\mathbf{e}|\mathbf{d})$ from a finite set of training samples of $[\mathbf{e}, \mathbf{d}]$. Of course this would require the extension of the deep neural network methodology to take the AVA-type data as input and the redefinition of the neural network as a 3-D recursive operator. What is more, this extended deep neural network methodology could be used to perform Bayesian seismic inversion in a single step (rather than the two-stage inversion assumed throughout this thesis): the neural network could be trained to emulate the mapping from the AVA-type data to the posterior over the elastic *and* geological parameters simultaneously, i.e., $\mathbf{d} \rightarrow p(\mathbf{g}, \mathbf{e}|\mathbf{d})$ from a finite set of training samples of $[\mathbf{e}, \mathbf{g}, \mathbf{d}]$.

Unfortunately, we were unable to successfully apply a deep neural network trained to take the AVA-type data as input or act as a 3-D (or even 2-D) recursive operator, in practice. Fundamentally, this is due to the increased number of neural network inputs required to redefine the neural network for these purposes. Extension to single-stage seismic inversion would only increase this problem since it would effectively increase the dimensionality of the data which must be processed by the neural network (i.e., we would have to train it to take $[\mathbf{e}, \mathbf{g}, \mathbf{d}]$ as input rather than just $[\mathbf{e}, \hat{\mathbf{e}}]$). Furthermore, if the aim of inversion were the geological parameters $\mathbf{g}$ then

it would no longer be acceptable for the neural network to predict a single value as output (i.e., the conditional expectation, as per the approximation in Chapter 2), since an estimate of uncertainty is usually required on any estimate of $\mathbf{g}$ (as argued in section 1.6). Instead, the neural network would have to predict the full posterior probability distribution $p(\mathbf{g}, \mathbf{e}|\mathbf{d})$ (as the mixture density network method does for the 'cell-wise' geological posteriors in Chapter 3). This would considerably increase the amount of training data (and hence the computational cost of training) required to determine the neural network. Thus, it seems that a realistic 'first step' for future work in this research area is to investigate how to modify the existing deep neural network method as described in Chapter 2 (i.e., a 1-D operator which predicts a single value, the conditional expectation, rather than the full posterior) to take the AVA-type data as input to predict an estimate of the elastic parameters (including high-fidelity prior information).

Chapter 7

Conclusion

In this thesis Bayesian seismic inversion and its computational costs were reviewed. The purpose of such inversions is to combine information from seismic data and prior geological knowledge to determine a posterior probability distribution over the elastic and geological parameters of the subsurface. Typically the subsurface is modelled by a cellular grid containing thousands or millions of cells within which these parameters are to be determined. Consequently the computational cost of determining the posterior distribution is usually very high. Thus in practice approximations to Bayesian seismic inversion must be considered. A particular, existing approximate workflow was described in this thesis: the so-called two-stage inversion method explicitly splits the problem into elastic and geological inversion stages. These two stages sequentially estimate the elastic parameters given the seismic data, and then the geological parameters given the elastic parameter estimates, respectively. In this thesis a number of methodologies were developed which enhance the accuracy of this approximate workflow.

Elastic inversion can be expensive since it involves inversion of the forward physics relating the elastic parameters to the seismic data. Thus the prior information (about the elastic parameters) employed is often simplified in order to reduce the computational cost of Bayesian inversion in this stage. Therefore a methodology was developed which efficiently transforms the results of such inversions (i.e., estimates constrained only by simple geological prior information) into new estimates containing sophisticated prior geological information. The transformation is performed by recursively applying a deep neural network function to individual traces of the elastic parameter estimates. The method was shown (by comparison to well-

log measurements) to improve the resolution and accuracy of real elastic parameter estimates made over a reservoir model. However, it was found that the accuracy of the results of the method were dependent upon those of the original elastic inversion. Thus in future the method should be extended to the direct inversion of the seismic data for estimates of the elastic parameters (containing sophisticated prior geological information), thus avoiding existing elastic inversion methods altogether.

It was described how so-called mixture density neural network (MDN) inversion may be used to solve the geological inversion problem analytically (and thus very rapidly and efficiently), but only if it is assumed that (i) there is no prior correlation between the geological parameters in different grid cells, and (ii) the marginal distributions over the geological parameters in each cell are identical. Thus a so-called prior replacement operation was developed which permits assumption (ii) to be relaxed, and hence increases the range of applicability of MDN inversion. The method was demonstrated for a synthetic geological inversion problem and was shown to be orders of magnitude faster than existing methods for varying the prior distribution in MDN inversion.

Furthermore, it was shown that prior replacement can be used to integrate the efficient MDN-derived solutions within general, stochastic geological inversion methods that are not restricted by assumption (i), above. Such general inversion methods use Markov-chain Monte-Carlo (MCMC) sampling, thus they estimate the posterior over the geological parameters by producing a correlated chain of samples from it. It was shown that this approach can yield biased estimates of this posterior. Thus an alternative method which obtains a set of non-correlated samples from the posterior was developed, avoiding the possibility of bias in the estimate. The method uses a recursive algorithm to calculate a set of conditional distributions which permit exact, non-correlated sampling from the posterior. The computational cost of the algorithm was shown to scale exponentially with the size of the model grid and the range of spatial dependency of the multi-point geostatistical model used to specify prior geological information. An approximate version of the algorithm was developed which could be applied to realistically-sized two-dimensional model grids. It was applied to a synthetic geological inversion problem for lithology-fluid class over such a grid. It compared well to the results of Gibbs sampling (a MCMC inversion method) which demonstrated quite severe bias, which was absent in the results of the recursive algorithm. However, the geostatistical model used had a relatively small range of spatial dependency. Thus, future work must focus on extending this

exact-sampling method to three-dimensional grids and to geostatistical models with a larger range of spatial dependency.

The prior geological information used in seismic inversion is codified within the geological prior probability distribution. It can be specified by a geostatistical model, parametrised by a set of statistics appropriate for the given application. These statistics can be derived from real images which bear similarity to the so-called target geology anticipated within the subsurface (that is, the expected spatial patterns within the subsurface geology, not its absolute distribution). Real training images are not always available from which these statistics may be extracted, in which case they may be generated by geological experts. However, this process can be costly and difficult. Thus an elicitation method was developed which obtains the appropriate statistics reliably and directly from a geological expert, without the need for training images. The method estimates the set of statistics which, when used within a given geostatistical model, generates realisations of the geological parameters which match the expert's mental envisagement of the target geology. The algorithm iteratively improves (using a genetic algorithm) a set of vectors of statistics, based on the input of the expert. It was demonstrated by providing 12 experts with a physical target image (geology), and prompting them to determine the corresponding statistics. The majority of experts were able to obtain a statistics vector which produced realisations which, to the best of their ability, had geology which was indistinguishable from that of the target image. Thus it was shown that the elicitation method may be used to determine the statistics used to specify a (geostatistical model, and hence a) geological prior distribution for seismic inversion. However, the speed of the algorithm is dependent upon the number of statistics to be determined, thus future work must focus on ensuring the applicability of the method to more sophisticated geostatistical models.

Overall a number of methods were developed which aimed to enhance existing Bayesian seismic inversion methodologies, particularly the two-stage inversion workflow. Additionally, in future the deep neural network methodology may offer a new method for seismic inversion; it was argued that it may be adapted to perform inversion in a single-step, by taking the seismic data as input and returning an estimate of the posterior distribution over the elastic and geological parameters, simultaneously. The methodologies developed in this thesis are quite general and may be applicable to a variety of (spatial) Bayesian inversion problems.

Appendix A

AVA forward model matrices

In order to be able to write the single matrix equation for the AVA forward model (equation 1.6), a single reflectivity vector is constructed by concatenating the reflectivity vectors for each of the angular ranges,

$$\mathbf{R}(\mathbf{e_x}) = \begin{pmatrix} \mathbf{r}_{near,\mathbf{x}}(\mathbf{e_x}) \\ \mathbf{r}_{mid,\mathbf{x}}(\mathbf{e_x}) \\ \mathbf{r}_{far,\mathbf{x}}(\mathbf{e_x}) \end{pmatrix}, \tag{A.1}$$

and a corresponding wavelet block-matrix by concatenating the wavelet Toeplitz matrices,

$$\mathbf{S} = \begin{pmatrix} \mathbf{s}_{near} & 0 & 0 \\ 0 & \mathbf{s}_{mid} & 0 \\ 0 & 0 & \mathbf{s}_{far} \end{pmatrix}. \tag{A.2}$$

Then the AVA-type data for the different angular ranges is arranged into a single vector as

$$\mathbf{d}(\mathbf{e_x}) = \begin{pmatrix} \mathfrak{d}_{near,\mathbf{x}}(\mathbf{e_x}) \\ \mathfrak{d}_{mid,\mathbf{x}}(\mathbf{e_x}) \\ \mathfrak{d}_{far,\mathbf{x}}(\mathbf{e_x}) \end{pmatrix}. \tag{A.3}$$

Appendix B

Back propagation

We seek to minimise the sum-of-squares error in equation 2.11 with respect to the values of the network's weights (stored in the matrix $\mathbf{W}$ where $W_{i,j,l} = w_{ij}^{l}$). We perform the minimisation using gradient descent thus we are required to calculate the derivatives of the error function with respect to the weights. For a given weight, this can be calculated by summation of each term in the sum in equation 2.11, differentiated with respect to the given weight. Each of the terms in the sum represent the error function for a single realisation of the input-output pair in the training dataset, and can be written for the p^{th} instance in the training dataset as

$$E_p = \sum_{i=1}^{K^L} (v_{pi}^s - a_i^L(\mathbf{u}_p^s; \mathbf{W}))^2, \tag{B.1}$$

where v_{pi}^s is the i^{th} element of the p^{th} output vector in the training dataset, a_i^L is the i^{th} output node (that is, node in layer L of the network) of the neural network and $\mathbf{u}_p^s$ is the (entire) p^{th} input vector in the training dataset. Note that in section 2.7.1 (and hence equation 2.11) we used the notation $\mathbf{q}(\mathbf{u}; \mathbf{W}) = \mathbf{a}^L(\mathbf{u}; \mathbf{W})$ to represent the output of the neural network, but here we use $\mathbf{a}^L = [a_1^L, ..., a_{K^L}^L]$ to specifically reference the individual variables in the output layer L (as in equation B.1).

The summation in equation B.1 is made over the K^L elements in the output vector (i.e., the number of nodes in the output layer of the network, discounting the redundant bias node). In order to derive the back-propagation expression for the derivative of equation B.1 with respect to a given weight, we begin by writing the derivative of the j^{th} variable in the l^{th} layer of the network, a_j^l, with respect to the i^{th} variable in the layer below, a_i^{l-1}. Given that we assume a sigmoidal activation

function in equation 2.10, this derivative is equal to

$$\frac{dz_j^l}{dz_i^{l-1}} = w_{ij}^l z_j^l \left(1 - a_j^l\right). \tag{B.2}$$

Similarly the derivative of a_j^l with respect to the i^{th} weight which connects directly to it from the layer below, w_{ij}^l, is given by

$$\frac{dz_j^l}{dw_{ij}^l} = a_i^{l-1} a_j^l \left(1 - a_j^l\right). \tag{B.3}$$

We can then calculate the derivative of the error function with respect to w_{ij}^l as

$$\frac{\partial E_p}{\partial w_{ij}^l} = \frac{\partial E_p}{\partial a_j^l} \frac{\partial a_j^l}{dw_{ij}^l}. \tag{B.4}$$

Now we define the so-called back-propagated error, denoted δ_i^l, as the derivative of equation B.1 with respect to the i^{th} variable in the l^{th} layer of the network, a_i^l, that is

$$\delta_i^l \equiv \frac{\partial E_p}{\partial a_i^l}, \tag{B.5}$$

thus for all *hidden* layers, δ_i^l can be written using the sum rule for partial derivatives as

$$\delta_i^l = \sum_{j=0}^{K^{l+1}} \frac{\partial E_p}{\partial a_j^{l+1}} \frac{\partial a_j^{l+1}}{\partial a_i^l} \quad \forall\, l \in\ [1, ..., L-1], \tag{B.6}$$

then substituting equation B.2 into this we have that

$$\delta_i^l = a_i^l \left(1 - a_i^l\right) \sum_{j=0}^{K^{l+1}} w_{ij}^l \frac{\partial E_p}{\partial a_j^{l+1}} \quad \forall\, l \in\ [1, ..., L-1]. \tag{B.7}$$

It is clear that this expression for the back-propagated error for a node in layer l, that is δ_i^l, contains the back-propagated error for all nodes in layer $l+1$, that is

$\delta_j^{l+1}\ \forall\ j$. Thus equation B.6 may be written as the recursive relation

$$\delta_i^l = a_i^l\left(1-a_i^l\right)\sum_{j=0}^{K^{l+1}} w_{ij}^l \delta_j^{l+1}\ \ \forall\ l \in\ [1,...,L-1]. \tag{B.8}$$

This recursion may be initiated by calculating the back-propagated error (using its definition in equation B.5) for the output layer, i.e., $\delta_i^{l=L}$, by directly differentiating the error function in equation B.1 with respect to the output variables as

$$\delta_i^{l=L} = \frac{\partial E_p}{\partial a_i^L} = 2\left(v_{pi}^s - a_i^L\right). \tag{B.9}$$

The recursively-calculated back-propagated error δ_i^l can be used within equation B.4 to calculate the derivative of the error function with respect to any weight. Thus the gradient used for gradient descent in the weight space can be calculated where each element is given by

$$\Delta w_{ij}^l = -\eta\frac{1}{N}\sum_{p=1}^{N}\frac{\partial E_p}{\partial w_{ij}^l} \tag{B.10}$$

where η, the so-called learning rate parameter, is a non-negative constant which controls the step-size of gradient descent.

Appendix C

Stacked denoising-autoencoder pre-training

Essentially, this method of pre-conditioning of $\mathbf{W}$ involves isolating layers of nodes and training them sequentially to encode and decode the input portion of the dataset. To do this, initially a new one-hidden-layer network is formed by setting its hidden layer of nodes to be equal to the layer of nodes $l = 1$, and both its output and input layers to be equal to the layer of nodes $l = 0$ (the input layer), in the original network. This so-called isolated network is then trained using gradient-descent (using back-propagation - see Appendix B) to encode and decode the input variable $\mathbf{u}$. To do this a separate training dataset is created comprising N pairs of $\mathbf{u}^s$ as both input and output.

After training, all N instances of the hidden layer variables $\mathbf{a}^1$, generated by supplying each of the N input vectors $\mathbf{u}^s$ in the training dataset to the isolated network, are calculated and retained. Then as for $l = 1$, a new isolated one-hidden-layer network is formed using the layer of nodes $l = 2$ as the hidden layer, and $l = 1$ as the input and output. Then this is trained to encode and decode the hidden layer of nodes (variables) $l = 1$, using the N encoded $\mathbf{a}^1$ instances to form the training dataset (that is, both its input and output). This process is then repeated for all layers until $l = L$.

The isolated networks may be termed autoencoders since their inputs and outputs are defined to be the same. However, so-called 'masking' noise is applied to their training datasets, which means that a certain percentage ϕ of the inputs in their individual (encoded) training datasets are set to zero. The use of this type of noise

is effective at encouraging autoencoders to recover noiseless versions of noisy input (van der Maaten et al., 2009). After each autoencoder is trained its weights are used to form part of the initial $\mathbf{W}$ matrix used in training of the network as a whole, that is for minimising equation 2.11 (with the input and output set equal to the training instances of the input $\mathbf{u}^s$ and output $\mathbf{v}^s$ variables, respectively). Full details of the stacked-denoising autoencoder pre-training procedure can be found in van der Maaten et al. (2009) or Vincent et al. (2010).

References

van der Maaten, L. J., E. O. Postma, and H. J. van den Herik (2009), Dimensionality reduction: A comparative review, *Journal of Machine Learning Research*, *10*(1-41), 66–71.

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol (2010), Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *The Journal of Machine Learning Research*, *9999*, 3371–3408.

Appendix D

Prior replacement in mixture density network inversion

D.1 Preliminaries

In this appendix we define the prior replacement equations for the output of mixture density network inversion in greater detail. For clarity we ignore the i subscript here thus $\mathbf{m}$ and $\mathbf{e}$ should be read as $\mathbf{m}_i$ and $\mathbf{e}_i$, respectively, in the following derivations.

We define two domains M_{old} and M_{new} which correspond to the non-zero regions of $p_{old}(\mathbf{m})$ and $p_{new}(\mathbf{m})$, respectively. As described in section 3.4, $p_{new}(\mathbf{m})$ must be zero everywhere that $p_{old}(\mathbf{m})$ is zero, thus

$$M_{new} \subseteq M_{old}. \tag{D.1}$$

In general the priors are referred to as $p_{new}(\mathbf{m})$ and $p_{old}(\mathbf{m})$. However, we will employ Uniform distributions frequently so it is useful to define a Uniform distribution for both of these now, to aid the analysis in the following sections. We define a boxcar-like function δ, which has the properties

$$\delta(\mathbf{m}; M) = \begin{cases} 0 & \text{for } \mathbf{m} \notin M \\ 1 & \text{for } \mathbf{m} \in M \end{cases} \tag{D.2}$$

where $\mathbf{m}$ is the model vector and M is a region of the space of possible $\mathbf{m}$'s. Thus

we define Uniform new and old priors for later use:

$$u_{old}(\mathbf{m}) = c_{old}\delta(\mathbf{m}; M_{old}) \tag{D.3}$$

$$u_{new}(\mathbf{m}) = c_{new}\delta(\mathbf{m}; M_{new}) \tag{D.4}$$

where the constants c_{old} and c_{new} are probability densities, whose exact values are related to the volumes of M_{old} and M_{new} (but are not important here).

D.2 Calculating the posterior PDF with a Uniform 'old' prior

If $M_{new} \subseteq M_{old}$ is true and $p_{old}(\mathbf{m}) = u_{old}(\mathbf{m})$ then equation 3.9 can be simplified because $p_{old}(\mathbf{m})$ is constant over the volume in which $p_{new}(\mathbf{m}) \neq 0$. Substituting equation D.3 into equation 3.5 we obtain

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k}\frac{p_{new}(\mathbf{m})}{p_{old}(\mathbf{m})}p_{old}(\mathbf{m}|\mathbf{d}) = \frac{1}{k}\frac{p_{new}(\mathbf{m})}{c_{old}\delta(\mathbf{m}; M_{old})}p_{old}(\mathbf{m}|\mathbf{d}). \tag{D.5}$$

Given that $M_{new} \subseteq M_{old}$, $p_{new}(\mathbf{m})$ has zero probability density throughout the extent of the region of zero probability density of $u_{old}(\mathbf{m})$. Therefore, if we stipulate that $\mathbf{m} \in M_{new}$, the box-car function is unnecessary and may be removed from equation D.5 thus:

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k}\frac{p_{new}(\mathbf{m})}{c_{old}}p_{old}(\mathbf{m}|\mathbf{d}), \quad m \in M_{new}. \tag{D.6}$$

Similarly, substituting equation D.3 into equation 3.6 we obtain

$$k = \int_{-\infty}^{+\infty} \frac{1}{k}\frac{p_{new}(\mathbf{m})}{c_{old}\delta(\mathbf{m}; M_{old})}p_{old}(\mathbf{m}|\mathbf{d})d\mathbf{m}, \tag{D.7}$$

and again stipulating that $\mathbf{m} \in M_{new}$ allows the boxcar function to be removed and the limits of integration to be set to M_{new} thus

$$k = \int_{M_{new}} \frac{p_{new}(\mathbf{m})}{c_{old}}p_{old}(\mathbf{m}|\mathbf{d})d\mathbf{m}. \tag{D.8}$$

Combining equations D.6 and D.8 and cancelling the constants we obtain the equation

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k'} p_{new}(\mathbf{m}) p_{old}(\mathbf{m}|\mathbf{d}) \tag{D.9}$$

where the normalising constant is

$$k' = \int_{M_{new}} p_{new}(\mathbf{m}) p_{old}(\mathbf{m}|\mathbf{d}). \tag{D.10}$$

It should be noted that the change in the limit of integration in equation D.8 may not be trivial if the dimensionality of the model space is high and/or the Uniform distribution has complicated bounds.

D.3 Calculating the posterior with a Uniform old prior and Uniform new prior

Equations D.9 and D.10 can be used under the conditions that $M_{new} \subseteq M_{old}$ and the old prior is Uniform, $p_{old}(\mathbf{m}) = u_{old}(\mathbf{m})$. If also the new prior is Uniform, $p_{new}(\mathbf{m}) = u_{new}(\mathbf{m})$, then the result is simpler. Combining equations D.4, D.9 and D.10 we obtain

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{c_{new}\delta(\mathbf{m}; M_{new}, p_{old})\, p_{old}(\mathbf{m}|\mathbf{d})}{c_{new}\int_{M_{new}} \delta(\mathbf{m}; M_{new})\, p_{old}(\mathbf{m}|\mathbf{d})\, d\mathbf{m}}, \tag{D.11}$$

As before $\mathbf{m} \in M_{new}$ so the boxcar functions may be removed, thus

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{\int_{M_{new}} p_{old}(\mathbf{m}|\mathbf{d}) d\mathbf{m}} p_{old}(\mathbf{m}|\mathbf{d}) \tag{D.12}$$

Recognising that we have now a normalising constant in the denominator, which we denote with k'', we rewrite equation D.12 as

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k''} p_{old}(\mathbf{m}|\mathbf{d}), \quad m \in M_{new} \tag{D.13}$$

Substituting equation 3.8 into D.13 yields

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k''} \sum_{j=1}^{K} \alpha_j \phi(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j), \quad m \in M_{new} \tag{D.14}$$

where

$$k'' = \int_{M_{new}} p_{old}(\mathbf{m}|\mathbf{d})d\mathbf{m} = \int_{M_{new}} \sum_{j=1}^{K} \alpha_j \phi(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j)d\mathbf{m}. \tag{D.15}$$

Evaluation of the normalising constant k'' requires only the integration of the series of Gaussians (the GMM) in equation D.15 over the non-zero region of M_{new}. This implies the need to evaluate a definite integral of a multivariate normal distribution. Whilst this does not have an analytic expression (Drezner, 1992), it has been widely studied due to its importance in probability theory. Many algorithms exist for its evaluation (Drezner and Wesolowsky, 1990; Genz and Bretz, 1999, 2002; Genz, 2004), apart from simple numerical integration techniques (Riley et al., 2006, pp. 1000-1009).

D.4 Calculating the posterior with Uniform old prior and Gaussian new prior

If $p_{old}(\mathbf{m})$ is Uniform and $p_{new}(\mathbf{m})$ is a Gaussian then we can use equation D.10 to evaluate the normalising constant in equation D.9, and hence find the new posterior. We must explicitly state that this new prior obeys $M_{new} \subseteq M_{old}$, that is that its non-zero extent is limited to that of the old prior. Thus, we define the new prior as a truncated Gaussian - the product of a Gaussian and the boxcar-type function defined in equation D.4:

$$p_{new}(\mathbf{m}) = c\phi(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new})\,\delta(\mathbf{m}; M_{new}) \tag{D.16}$$

where c is a (normalising) constant. We use the notation $\phi(\mathbf{m}; \mu, \mathbf{\Sigma})$ to denote a normalised Gaussian function as a function of $\mathbf{m}$ with mean vector μ and covariance matrix $\mathbf{\Sigma}$. The subscript new indicates that we refer to parameters belonging to the new prior, p_{new}. Substituting equations 3.8 and D.16 into equation D.9, the c constant disappears henceforth (since it exists in both the numerator and denominator), then stipulating that $\mathbf{m} \in M_{new}$ allows us to write

$$p_{new}(\mathbf{m}|\mathbf{d}) = \frac{1}{k'}\phi(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new}) \sum_{j=1}^{K} \alpha_j \phi(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j), \quad m \in M_{new}. \tag{D.17}$$

Similarly, for the normalising constant we can substitute equations D.16 and 3.8 into equation D.10, and since $\mathbf{m} \in M_{new}$ remove the boxcar function, thus

$$k' = \int_{M_{new}} \phi(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new}) \sum_{j=1}^{K} \alpha_j \phi(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j) d\mathbf{m}. \tag{D.18}$$

In order to simplify equation D.18 and subsequently to evaluate equation D.17 we use the result that the product of two Gaussians is an un-normalised Gaussian (Ahrendt, 2005). This allows us to obtain an analytical expression for a series of single Gaussians within each of these equations. We can combine the Gaussians as such (Ahrendt, 2005)

$$\sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j\right) \phi\left(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new}\right) = \sum_{j=1}^{K} \alpha_j R_j \phi\left(\mathbf{m}; {\mu_j}', {\mathbf{\Sigma}_j}'\right) \tag{D.19}$$

where the mean and covariance parameters are now given by

$${\mu'}_j = \left({\mathbf{\Sigma}'}_j {\mathbf{\Sigma}_{new}}^{-1} \mu_{new}\right) + \left({\mathbf{\Sigma}'}_j {\mathbf{\Sigma}_j}^{-1} \mu_j\right) \text{ and } {\mathbf{\Sigma}'}_j = \left({\mathbf{\Sigma}_{new}}^{-1} + {\mathbf{\Sigma}_j}^{-1}\right)^{-1}, \tag{D.20}$$

and the constant R_j is given by

$$R_j = \left|2\pi \left(\mathbf{\Sigma}_{new} + \mathbf{\Sigma}_j\right)\right|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mu_{new} - \mu_j)^T \left(\mathbf{\Sigma}_{new} + \mathbf{\Sigma}_j\right)^{-1} \left(\mu_{new} - \mu_j\right)\right]. \tag{D.21}$$

Upon substitution of the Gaussian product given in equation D.19, equation D.17 becomes

$$p_{new}\left(\mathbf{m}|\mathbf{d}\right) = \frac{1}{k'} \sum_{j=1}^{K} \alpha_j R_j \phi\left(\mathbf{m}; {\mu_j}', {\mathbf{\Sigma}_j}'\right) \tag{D.22}$$

and equation D.18 becomes

$$k' = \int_{M_{new}} \sum_{j=1}^{K} \alpha_j R_j \phi\left(\mathbf{m}; {\mu_j}', {\mathbf{\Sigma}_j}'\right) d\mathbf{m}. \tag{D.23}$$

Equation D.23 can be evaluated by integration over the truncated Gaussians as in the previous section. Once this is substituted into equation D.22 the full posterior can be calculated.

D.5 Calculating the posterior with both old and new Gaussian priors

The special case of having both a Gaussian old prior $p_{old}(\mathbf{m})$, and a Gaussian new prior $p_{new}(\mathbf{m})$, is interesting since this may permit the normalisation constant to be calculated analytically in equations 3.9 and 3.10. To see this we explicitly expand the priors in terms of Gaussian kernels. In contrast to the previous section, we express the new and old priors as full Gaussians so we do not need to truncate either prior as they both span the infinite model space. Therefore

$$p_{old}(\mathbf{m}) = \phi(\mathbf{m}; \mu_{old}, \mathbf{\Sigma}_{old}), \tag{D.24}$$

and

$$p_{new}(\mathbf{m}) = \phi(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new}). \tag{D.25}$$

Since both priors are Gaussian we substitute equations D.24 and D.25 into equation 3.10,

$$k = \int_{-\infty}^{+\infty} \frac{\phi(\mathbf{m}; \mu_{new}, \mathbf{\Sigma}_{new}) \sum_{j=1}^{K} \alpha_j \phi(\mathbf{m}; \mu_j, \mathbf{\Sigma}_j)}{\phi(\mathbf{m}; \mu_{old}, \mathbf{\Sigma}_{old})} d\mathbf{m}. \tag{D.26}$$

As previously, the Gaussians can be combined in some way to make the calculation simpler. There are two ways of combining the Gaussians in equation D.26. We could divide the GMM by the old prior and then multiply by the new prior, or we could divide the new prior by the old prior and then multiply by the GMM. We discuss the latter here as it is much simpler because it involves only the division of two single Gaussians rather than involving the series of Gaussians in the division (since this is more complicated than the multiplication of two Gaussians, as discussed below).

The multiplication of one Gaussian by another is always Gaussian (Bromiley, 2003), therefore if we can ensure that the division of the new prior by the old prior is Gaussian then the whole operation will always yield a Gaussian. However, the division of one Gaussian by another does not always yield a Gaussian. This can be seen by first writing out the expression for a multivariate Gaussian

$$\phi(\mathbf{m}; \mu, \mathbf{\Sigma}) = (2\pi)^{-\frac{n}{2}} |\mathbf{\Sigma}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{m}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{m}-\mu)} \tag{D.27}$$

where n is the dimensionality of $\mathbf{m}$. For the expression in equation D.27 to behave as a Gaussian the covariance matrix must be positive definite (Rue and Held, 2005).

Then, since the inverse of a positive definite matrix is positive definite, the condition

$$\mathbf{m}^T\boldsymbol{\Sigma}^{-1}\mathbf{m} > 0 \quad \forall\, \mathbf{m} \in \mathbb{R}^d \tag{D.28}$$

must be true for a valid Gaussian. We can write the division of the new by the old prior in equation D.26 as a product but with the covariance matrix of the old prior multiplied by -1, thus

$$k = \int_{-\infty}^{+\infty} \phi\left(\mathbf{m}; \mu_{new}, \boldsymbol{\Sigma}_{new}\right) \phi\left(\mathbf{m}; \mu_{old}, -\boldsymbol{\Sigma}_{old}\right) \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}; \mu_j, \boldsymbol{\Sigma}_j\right) d\mathbf{m} \tag{D.29}$$

and the Gaussian division within this can be written in the form of a single Gaussian, i.e., $\phi\left(\mathbf{m}; \mu_{new}, \boldsymbol{\Sigma}_{new}\right) \phi\left(\mathbf{m}; \mu_{old}, -\boldsymbol{\Sigma}_{old}\right) = \phi\left(\mathbf{m}; \mu', \boldsymbol{\Sigma}'\right)$, thus we now have

$$k = \int_{-\infty}^{+\infty} \phi\left(\mathbf{m}; \mu', \boldsymbol{\Sigma}'\right) \sum_{j=1}^{K} \alpha_j \phi\left(\mathbf{m}; \mu_j, \boldsymbol{\Sigma}_j\right) d\mathbf{m}. \tag{D.30}$$

The equations for the mean vector, covariance matrix and normalisation constant (given in equations D.20 and D.21) for the product of two Gaussians are then valid for $\phi\left(\mathbf{m}; \mu', \boldsymbol{\Sigma}'\right)$. Thus for the result of the Gaussian division, from equation D.20 we have

$$\boldsymbol{\Sigma}' = \left(\boldsymbol{\Sigma}_{new}^{-1} - {\boldsymbol{\Sigma}_{old}}^{-1}\right)^{-1}, \tag{D.31}$$

and

$$\mu' = \left(\boldsymbol{\Sigma}'\boldsymbol{\Sigma}_{new}^{-1}\mu_{new}\right) - \left(\boldsymbol{\Sigma}'\boldsymbol{\Sigma}_{old}^{-1}\mu_{old}\right). \tag{D.32}$$

Clearly, $\boldsymbol{\Sigma}'$ must be positive definite for the Gaussian division to yield a valid Gaussian. In other words, the condition in equation D.28 must apply to $\boldsymbol{\Sigma}'$. Thus substituting equation D.31 into D.28 yields

$$\mathbf{m}^T{\boldsymbol{\Sigma}'}^{-1}\mathbf{m} = \mathbf{m}^T\left(\boldsymbol{\Sigma}_{new}^{-1} - \boldsymbol{\Sigma}_{old}^{-1}\right)\mathbf{m} > 0 \quad \forall\, \mathbf{m} \in \mathbb{R}^d \tag{D.33}$$

which may be rewritten to give the condition as

$$\mathbf{m}^T\boldsymbol{\Sigma}_{new}^{-1}\mathbf{m} - \mathbf{m}^T\boldsymbol{\Sigma}_{old}^{-1}\mathbf{m} > 0 \quad \forall\, \mathbf{m} \in \mathbb{R}^d. \tag{D.34}$$

If both the old and new priors are valid Gaussians then their covariance matrices are positive definite and obey equation D.28. Thus equation D.34 cannot be true for all

possible $\mathbf{\Sigma}_{new}$ and $\mathbf{\Sigma}_{old}$. In order to ensure that equation D.33 holds we could design the new and old priors specifically by manipulating their eigen-decompositions, for example (but we will not discuss such possibilities here). Usefully, if equation D.33 is true, equation D.32 will always give a valid (i.e., real) mean vector for the resulting Gaussian. Therefore, the values of the mean vectors of the old and new priors do not effect whether the division of these two Gaussians yields another Gaussian or not, and so the means of the old and new priors may have any value.

References

Ahrendt, P. (2005), The multivariate gaussian probability distribution, *Tech. rep.*, Technical University of Denmark.

Bromiley, P. A. (2003), Products and convolutions of Gaussian distributions, *Tech. rep.*, University of Manchester.

Drezner, Z. (1992), Computation of the multivariate normal integral, *ACM Transactions on Mathematical Software (TOMS)*, *18*(4), 470–480.

Drezner, Z., and G. O. Wesolowsky (1990), On the computation of the bivariate normal integral, *Journal of Statistical Computation and Simulation*, *35*(1-2), 101–107.

Genz, A. (2004), Numerical computation of rectangular bivariate and trivariate normal and t probabilities, *Statistics and Computing*, *14*(3), 251–260.

Genz, A., and F. Bretz (1999), Numerical computation of multivariate t-probabilities with application to power calculation of multiple contrasts, *Journal of Statistical Computation and Simulation*, *63*(4), 103–117.

Genz, A., and F. Bretz (2002), Comparison of methods for the computation of multivariate t probabilities, *Journal of Computational and Graphical Statistics*, *11*(4), 950–971.

Riley, K., M. Hobson, and S. Bence (2006), *Mathematical methods for physics and engineering*, Cambridge University Press.

Rue, H., and L. Held (2005), *Gaussian Markov random fields: theory and applications*, Chapman & Hall.

Appendix E

Yin-Marion model

E.1 Yin-Marion shaly-sand model

The forward petrophysical model which we use is the Yin-Marion shaly-sand model (Marion, 1990; Yin et al., 1993; Avseth et al., 2005). In this model two distinct domains are defined for sand-shale mixtures: sandstones with a secondary shale component, called shaly-sands, and shales with secondary sand component, called sandy-shales. In the former domain clay particles are assumed to be within the pore space of a sandstone frame. Increasing shale content fills this pore space, decreasing porosity linearly. Thus in this case the porosity varies according to

$$\phi = \phi_s - C\left(1 - \phi_{sh}\right),\ \forall\, C < \phi_s \tag{E.1}$$

where C is the shale volume fraction, ϕ_s is porosity of the clean sandstone frame and ϕ_{sh} is the intrinsic porosity of the shale. In the other domain, the sandy-shale domain, the shale volume fraction is greater than the porosity of the clean sandstone frame. In this case the rock is no longer considered to consist of a sandstone frame with a pore space, but instead it is considered to be shale with sand inclusions. There is no sandstone porosity, only isolated grains, and the only porosity which exists is within the intrinsic pore space of the shale. The total porosity is then:

$$\phi = C\phi_{sh},\ \forall\, C \geq \phi_s. \tag{E.2}$$

The volume fractions of the components (i.e., shale, sand and pore fluid) predicted by these equations can then be treated in a number of different ways to predict

the S-wave impedance I_S, and P-wave impedance I_P, of the bulk rock. To do this, we chose to use the upper Hashin-Shtrikman bound for the mixture in the shaly-sand case and the lower bound in the sandy-shale case (Avseth et al., 2005) to approximately simulate the two different assumed micro-geometries of the domains (see Mavko et al. (2009), for an explanation of the micro-geometry implied by these bounds). The densities can be calculated with the volume fractions and the known densities of the constituents. We assumed a constant mineralogy of the shale and sand components in this model. However, we assumed that the pore fluid consisted of a water and a gas phase so a third model parameter is introduced: the water saturation, $S_{wt} \in [0, 1]$. The elastic moduli and densities of the shale, sand and pore-fluid (mixture of gas and water) could be taken from examples in the literature (e.g., Mavko et al., 2009). Note that the intrinsic porosity of shale is kept constant so in total only three model parameters could vary and we write the rock-physical parameter vector, at a cell in a subsurface model, as $\mathbf{m}_i = [m_1, m_2, m_3] = [C, \phi_s, S_{wt}]$.

E.2 The probabilistic forward model

We symbolically write the Yin-Marion shaley-sand model described above as $\mathbf{f}(\mathbf{m}_i)$. By definition, it is a deterministic model for predicting I_S and I_P given $\mathbf{m}_i$, but we included a random element by adding random Gaussian noise ($\mathbf{n}$) to its output. Thus the full uncertain forward model is written

$$\mathbf{e}_i = \mathbf{f}(\mathbf{m}_i) + \mathbf{n}, \quad \mathbf{n} \sim \phi\left(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{e}_i}\right), \quad \mathbf{\Sigma}_{\mathbf{e}_i} = \begin{bmatrix} \sigma_P^2 & 0 \\ 0 & \sigma_S^2 \end{bmatrix} \tag{E.3}$$

where $\mathbf{f}(\mathbf{m}_i)$ represents the Yin-Marion shaley-sand model, $\phi()$ has its usual meaning as a Gaussian function, $\mathbf{m}_i = [m_2, m_1]$ is the vector of model (i.e., continuous geological) parameters and $\mathbf{e}_i = [I_P, I_S]_i$ is the elastic parameter (impedances) vector. The random Gaussian noise is uncorrelated between I_S and I_P, and is specified by the standard deviation of error on I_P, $\sigma_P = 1.5 \times 10^4 s^{-1} m^{-2} kg$ and on I_S, $\sigma_S = 1.0 \times 10^4 s^{-1} m^{-2} kg$. Since the noise is Gaussian, an appropriate PDF describing the probability of the elastic parameters, given the model parameters at cell i, is written

$$p(\mathbf{e}_i|\mathbf{m}_i) = \frac{|\mathbf{\Sigma}_{\mathbf{e}_i}|^{-\frac{1}{2}}}{2\pi} \exp\left(-\frac{1}{2} - (\mathbf{e}_i - \mathbf{f}(\mathbf{m}_i))^T \mathbf{\Sigma}_{\mathbf{e}_i}{}^{-1} (\mathbf{e}_i - \mathbf{f}(\mathbf{m}_i))\right) \tag{E.4}$$

References

Avseth, P., T. Mukerji, and G. Mavko (2005), *Quantitative seismic interpretation*, Cambridge University Press.

Marion, D. P. (1990), Acoustical, mechanical, and transport properties of sediments and granular materials, Ph.D. thesis, Stanford University, Department of Geophysics.

Mavko, G., T. Mukerji, and J. Dvorkin (2009), *The rock physics handbook: Tools for seismic analysis of porous media*, Cambridge University Press.

Yin, H., A. Nur, and G. Mavko (1993), Critical porosity: A physical boundary in poroelasticity, in *International journal of rock mechanics and mining sciences & geomechanics abstracts,* pp.805–808, Pergamon.

Appendix F

Quality in the results of prior replacement

F.1 Quality of the posterior estimates from prior replacement

In the results obtained for single MDN inversions in section 3.7.1 (Figures 3.2 and 3.3) we observed qualitatively that prior replacement may out-perform prior-specific training in some aspects of the quality of the estimated posterior distribution. We hypothesised in section 3.8.2 that this effect could be attributed to the difference in the distribution of samples used to train the network in each case. We now test this hypothesis by investigating a Bayesian inverse problem in which *sampling* (rather than a neural network) is used to estimate a single posterior PDF.

To do this we suppose that we have a likelihood distribution which we can only evaluate up to a multiplicative constant, and a prior which we know parametrically. Consequently, we do not know the posterior (equation 3.3) analytically (i.e., we do not know the normalising constant - as is often the case in practical problems). The usual approach to such problems (Mosegaard and Sambridge, 2002) is to sample directly from the posterior using Monte-Carlo (MC) methods in order to estimate the posterior density. We call this *direct estimation*. However, since the prior is known analytically, the posterior can also be estimated by prior replacement. To do this we would construct an old posterior using the appropriate likelihood (i.e., that used in direct estimation) and a broad old prior (see equation 3.11, for example). We would then sample from this old posterior and estimate its density. Then the

prior replacement equations would be applied to replace the old prior with the new prior (i.e., the appropriate prior used in direct estimation). Henceforth we refer to this as *indirect estimation*.

Direct and indirect estimation are equivalent to prior-specific training and prior replacement, respectively, in the discussion of MDN inversion in Chapter 3. The only difference now is that we assume that the samples are being used to directly estimate a posterior for a given datum, rather than to estimate the parameters of a neural network which will predict the posterior for any data.

Henceforth, we analyse the quality of the posterior estimate obtained using direct and indirect estimation for a single continuous model parameter, m. For simplicity, we also assume that the data vector consists of only one element, thus the data in this 'toy' inverse problem is written d. Furthermore we assume that the forward function is such that it describes an unnormalised Gaussian over m (for an example of such a likelihood function, see Tarantola (2002, pp. 64-68)). This likelihood may be written as the product of a normalised Gaussian and a constant,

$$p\left(d|m\right) = c_1 \phi\left(m; \mu_L, \Sigma_L\right). \tag{F.1}$$

We assume also that the new prior is Gaussian, thus

$$p_{new}\left(m\right) = \phi\left(m; \mu_B, \Sigma_B\right). \tag{F.2}$$

The new posterior can then be formed by substituting equations F.1 and F.2 into equation 3.3. Cancelling the c_1 constants from the denominator and numerator of the resulting expression we obtain the normalised product of two Gaussians, which is another Gaussian

$$\begin{aligned} p_{new}\left(m|d\right) &= \frac{\phi\left(m; \mu_B, \Sigma_B\right) \phi\left(m; \mu_L, \Sigma_L\right)}{\int\limits_m \phi\left(m; \mu_B, \Sigma_B\right) \phi\left(m; \mu_L, \Sigma_L\right)} \\ &= \phi\left(m; \mu_P, \Sigma_P\right), \end{aligned} \tag{F.3}$$

where the new posterior Gaussian will have mean and variance given by

$$\Sigma_P = \left({\Sigma_B}^{-1} + {\Sigma_L}^{-1}\right)^{-1}, \quad \mu_P = \Sigma_P \left({\Sigma_B}^{-1}\mu_B + {\Sigma_L}^{-1}\mu_L\right) \tag{F.4}$$

(Bromiley, 2003). It should be noted that generally if we assume Gaussian forms for

our prior, likelihood and hence posterior there is no need for Monte-Carlo sampling and PDF estimation. However, we use this toy problem to investigate the difference between direct estimation (prior-specific training) and indirect estimation (prior replacement). We now describe direct and indirect estimation in more detail, and then also the methods by which we can compare the quality of the posterior estimates we obtain in each case.

F.1.1 Direct estimation

In direct estimation a set of N samples, $M_1, ..., M_i, ..., M_N$, are made directly from the new posterior, i.e., $M_i \sim p_{new}(m|d)$. These are then used to estimate the parameters of the new posterior distribution. We denote the estimate

$$\hat{p}_D(m|d) = \phi\left(m; \hat{\mu}_{P_D}, \hat{\Sigma}_{P_D}\right) \approx p_{new}(m|d) \tag{F.5}$$

where the maximum likelihood estimates (MLE) of the mean and variance are related to the N samples by

$$\hat{\mu}_{P_D} = \frac{1}{N}\sum_{i}^{N} M_i, \quad \hat{\Sigma}_{P_D} = \frac{1}{N-1}\sum_{i}^{N}(M_i - \hat{\mu}_{P_D})^2, \tag{F.6}$$

and these are therefore termed the direct estimators.

F.1.2 Indirect estimation

In indirect estimation samples are made from an old posterior and are used to estimate that distribution. Then prior replacement is used to determine an estimate of the new posterior by emplacing the appropriate new prior. Initially, we assume an infinitely-broad, Uniform old prior thus the PDF is constant (and improper, see e.g., Hobert and Casella (1996); Daniels (1999); Sun et al. (2001))

$$p_{old}(m) = c_2. \tag{F.7}$$

This is then used to construct the old posterior: by substituting equations F.1 and F.7 into equation 3.2, and cancelling constant terms we obtain a Gaussian

$$p_{old}(m|d) = \frac{c_1 c_2 \phi(m; \mu_L, \Sigma_L)}{\int_m c_1 c_2 \phi(m; \mu_L, \Sigma_L)\, dm} = \phi(m; \mu_L, \Sigma_L). \tag{F.8}$$

This is simply a normalised version of the likelihood. As in direct estimation, we then use N samples from this distribution to obtain an approximation to it, which we denote

$$\hat{p}_{old}(m|d) = \phi\left(m; \hat{\mu}_L, \hat{\Sigma}_L\right) \approx p_{old}(m|d). \tag{F.9}$$

The MLE estimators for the variance and mean parameters are given by

$$\hat{\mu}_L = \frac{1}{N}\sum_i^N M_i, \quad \hat{\Sigma}_L = \frac{1}{N-1}\sum_i^N (M_i - \hat{\mu}_L)^2 \tag{F.10}$$

where M_i now represents a set of N samples made from $\hat{p}_{old}(m|d)$. We now perform prior replacement in order to obtain an estimate of $p_{new}(m|d)$. To do this we substitute the expressions for the approximate old posterior, the old prior and the new prior (equations F.9, F.7 and F.2 respectively) into equation 3.5 such that we obtain an approximation for the new posterior given by

$$\hat{p}_I(m|d) = \frac{1}{k}\frac{\phi(m; \mu_B, \Sigma_B)}{c_2}\phi\left(m; \hat{\mu}_L, \hat{\Sigma}_L\right) \approx p_{new}(m|d) \tag{F.11}$$

where $\hat{p}_I(m|d)$ is used to denote this (indirect) approximation to $p_{new}(m|d)$. Making the same substitutions in equation 3.6 yields the approximate normalising constant

$$k \approx \int_{-\infty}^{+\infty} \frac{\phi(m; \mu_B, \Sigma_B)}{c_2}\phi\left(m; \hat{\mu}_L, \hat{\Sigma}_L\right) dm. \tag{F.12}$$

Substituting equation F.12 into equation F.11 and cancelling the constant old prior, c_2, we obtain

$$\begin{aligned}\hat{p}_I(m|d) &= \frac{\phi(m;\mu_B,\Sigma_B)\,\phi\left(m;\hat{\mu}_L,\hat{\Sigma}_L\right)}{\int_{-\infty}^{\infty}\phi(m;\mu_B,\Sigma_B)\,\phi\left(m;\hat{\mu}_L,\hat{\Sigma}_L\right)dm} \\ &= \phi\left(m;\hat{\mu}_{P_I},\hat{\Sigma}_{P_I}\right), \end{aligned} \tag{F.13}$$

which we have recognised as a normalised product of two Gaussians, which is a Gaussian. As such we can obtain the mean and variance using the standard identities for a Gaussian multiplication (Bromiley, 2003) as

$$\hat{\Sigma}_{P_I} = \left({\Sigma_B}^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1}, \quad \hat{\mu}_{P_I} = \hat{\Sigma}_{P_I}\left({\Sigma_B}^{-1}\mu_B + \hat{\Sigma}_L^{-1}\hat{\mu}_L\right), \tag{F.14}$$

and these are therefore termed the indirect estimators.

F.2 Comparing quality

To compare the quality of the two posterior estimates we calculate the variance and bias of the estimators (the mean and variance parameters) in each case. If we use the example of the variance parameter Σ, and the estimator of it $\hat{\Sigma}$, then the bias and the variance of the estimator are defined as

$$\text{bias}\left(\hat{\Sigma}\right) = \text{E}\left[\hat{\Sigma}\right] - \Sigma, \tag{F.15}$$

$$\text{var}\left(\hat{\Sigma}\right) = \text{E}\left[\left(\text{E}\left[\hat{\Sigma}\right] - \hat{\Sigma}\right)^2\right]. \tag{F.16}$$

Exact analytical expressions exist for these quantities for given N in the case of the direct estimators: they are simply those for a Gaussian which are well known (Ulrych et al., 2001, e.g.,), thus the biases are

$$\text{bias}\left(\hat{\mu}_{P_D}\right) = \text{bias}\left(\hat{\Sigma}_{P_D}\right) = 0. \tag{F.17}$$

and the variances are

$$\text{var}\left(\hat{\mu}_{P_D}\right) = \frac{\Sigma_P}{N}, \tag{F.18}$$

$$\mathrm{var}\left(\hat{\Sigma}_{P_D}\right) = \frac{2{\Sigma_P}^2}{N-1}. \tag{F.19}$$

No such exact analytical expressions exist for the bias and variance for the indirect estimators. However, we have derived approximations to these in Appendix G based on third-order Taylor expansions taken about the expected values of the $\hat{\mu}_L$ and $\hat{\Sigma}_L$ estimators (Oehlert, 1992; Van der Vaart, 2000):

$$\mathrm{bias}\left(\hat{\mu}_{P_I}\right) \approx \frac{1}{N-1}\left(\mu_L - \mu_P\right)\left(\Sigma_L^{-1}\Sigma_P - \Sigma_L^{-2}\Sigma_P^2\right), \tag{F.20}$$

$$\mathrm{bias}\left(\hat{\Sigma}_{P_I}\right) \approx \frac{2}{N-1}\left(\Sigma_L^{-2}{\Sigma_P}^3 - \Sigma_L^{-1}{\Sigma_P}^2\right) \tag{F.21}$$

and the variances are

$$\mathrm{var}\left(\hat{\mu}_{P_I}\right) \approx \left(\mu_P - \mu_L\right)^2 \frac{2\Sigma_P^2\Sigma_L^{-2}}{N-1} + \frac{\Sigma_P^2\Sigma_L^{-1}}{N}, \tag{F.22}$$

$$\mathrm{var}\left(\hat{\Sigma}_{P_I}\right) \approx \frac{2}{N-1}\frac{\Sigma_P^4}{\Sigma_L^2}. \tag{F.23}$$

Another measure of approximation quality is the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951), which measures the difference between two PDFs. Suppose that we make an estimate $\hat{p}(m|d)$ of a distribution $p(m|d)$. The KL divergence between the two, $D_{KL}\left[p(m|d) \,||\, \hat{p}(m|d)\right]$, is given by

$$D_{KL}\left[p\left(m|d\right) \,||\, \hat{p}\left(m|d\right)\right] = \int_{-\infty}^{+\infty} \ln\left(\frac{p(m|d)}{\hat{p}(m|d)}\right) p(m|d)\, dm. \tag{F.24}$$

This quantity is used extensively to measure approximation quality because of its intuitively appealing interpretation as the amount of information lost when approximating $p(m|d)$ by $\hat{p}(m|d)$ (Hershey and Olsen, 2007). Thus we interpret the KL divergence as a measure of the overall 'goodness of fit' of an approximate distribution. However, the advantage of the bias and variance quantities is that they are expected measures of the accuracy and precision, respectively, given a certain number of samples N. The KL divergence is only defined between two known distributions, therefore what we require is the expected KL divergence given that $\hat{p}(m|d)$ has been estimated using a certain number of samples N. No analytical expression exists for this quantity, thus we have to obtain an estimate of it empirically. That is to say, we must make a large number, L, of new posterior estimates and use this population to

estimate the average value, which would be calculated from the L estimates as

$$\mathrm{E}\left[D_{KL}\left[p\left(m|d\right)\ ||\,\hat{p}\left(m|d\right)\right]\right] \approx \frac{1}{L}\sum_{l}^{L} D_{KL}\left[p\left(m|d\right)\ ||\,\hat{p}_l\left(m|d\right)\right] \tag{F.25}$$

where $\hat{p}_l(m|d)$ is the l^{th} estimate of the posterior. Thus in practice we made an estimate of the posterior L times using both methods and calculated

$$D_{KL}\left[p(m|d)\,||\,\hat{p}_{I,l}(m|d)\right] \text{ and } D_{KL}\left[p(m|d)\,||\,\hat{p}_{D,l}(m|d)\right]$$

each time. Then from these two sets of L KL divergences we could calculate

$$\mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_I(m|d)\right]\right] \text{ and } \mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_D(m|d)\right]\right].$$

The number of estimates of the posterior we made in each case was $L = 1 \times 10^4$, whilst the number of samples made in each method was chosen to be $N = 10$. The analytical quantities (equations F.17 to F.23) can be calculated without any actual sampling. However, they do still require that the number of samples be specified. Thus when calculating these we chose $N = 10$ in both direct and indirect estimation for consistency.

It is clear that the relative properties of the old posterior (that is, the likelihood) and the new prior may effect the quality of the approximation derived by each method. Thus we do not calculate the quantities described above for just a single set of new and old posteriors; instead we vary these distributions systematically. Thus we repeated the above whilst varying the likelihood's parameters (the prior was kept constant since we are only interested in investigating the effect of the relative relationship of new prior and likelihood). We first investigated the effect of μ_L in isolation. To do this μ_L was varied and Σ_L kept constant. Secondly, we investigated the effect of Σ_L in isolation, by varying Σ_L and keeping μ_L constant. The results are described below.

F.3 Results

Firstly we varied μ_L in the range $[0, 4]$ at intervals of 0.1. The variance of the likelihood was kept constant at $\Sigma_L = 0.5$. The prior distribution was fixed with $\mu_B = 2$ and $\Sigma_B = 0.75$. This defined 41 different new posterior distributions, two examples of

which are plotted in Figure F.1(a) with the prior and likelihood distributions. The approximate expected Kullback-Leibler divergences for each of these scenarios for both methods, $\mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_I(m|d)\right]\right]$ and $\mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_D(m|d)\right]\right]$ are plotted in Figure F.1(b). The analytically calculated variance and bias of the estimators for both methods ($\hat{\Sigma}_{P_D}$, $\hat{\mu}_{P_D}$, $\hat{\Sigma}_{P_I}$ and $\hat{\mu}_{P_I}$) are plotted for comparison in Figure F.1 (c)-(f).

We then carried out exactly the same procedure except varying Σ_L rather than μ_L. Σ_L was varied in the range [0 4] at intervals of 0.1. The mean of the likelihood was kept constant at $\mu_L = 2$. The prior in this case had parameters $\mu_B = 4$ and $\Sigma_B = 1$. Again this defined 41 different new posterior distributions, two of which are plotted in Figure F.2(a) with the prior and likelihood distributions. The approximate expected Kullback-Leibler divergences for each of these scenarios, $\mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_I(m|d)\right]\right]$ and $\mathrm{E}\left[D_{KL}\left[p(m|d)\,||\,\hat{p}_D(m|d)\right]\right]$ are plotted in Figure F.2(b). The analytically calculated variance and bias of the estimators ($\hat{\Sigma}_{P_D}$, $\hat{\mu}_{P_D}$, $\hat{\Sigma}_{P_I}$ and $\hat{\mu}_{P_I}$) are plotted for comparison in Figure F.2 (c)-(f).

F.4 Interpretation of quality comparison results

We can make useful observations about the relative values of the variance and bias of the estimators $\hat{\Sigma}_{P_D}$, $\hat{\mu}_{P_D}$, $\hat{\Sigma}_{P_I}$ and $\hat{\mu}_{P_I}$ from their analytical expressions in equations F.17-F.19 and F.20-F.23 and the results in Figures F.1 and F.2.

When comparing equation F.17 to equation F.20, we see that $|\mathrm{bias}\left(\hat{\mu}_{P_I}\right)| > |\mathrm{bias}\left(\hat{\mu}_{P_D}\right)|$. However, $\mathrm{bias}\left(\hat{\mu}_{P_I}\right)$ will be zero in two non-trivial cases: where either (i) $\mu_P = \mu_L$, or (ii) $\Sigma_P = \Sigma_L$. From equation F.4 we can see that the former case implies that $\mu_B = \mu_L$, and that the latter case implies $\Sigma_B = \infty$ (i.e., the prior is flat). In Figures F.1(e) and F.2(e) we observe the bias of those estimators behaving in this way.

Inspecting equations F.18 and F.22 we see that it is possible that $\mathrm{var}\left(\hat{\mu}_{P_I}\right) < \mathrm{var}\left(\hat{\mu}_{P_D}\right)$, and that this will tend to be the case where (i) $\mu_P \to \mu_L$ or (ii) $\Sigma_L >> \Sigma_P$. Again from equation F.4 we can see that the former case implies that $\mu_B \to \mu_L$, and that the latter case implies $\Sigma_L >> \Sigma_B$. We can observe this behaviour in Figures F.1(f) and F.2(f).

Similarly, we see that generally $\mathrm{bias}\left(\hat{\Sigma}_{P_I}\right) > \mathrm{bias}\left(\hat{\Sigma}_{P_D}\right)$, and the only non-trivial exception to this is when $\Sigma_L = \Sigma_P$, where $\mathrm{bias}\left(\hat{\Sigma}_{P_I}\right) = 0$. Such behaviour can be

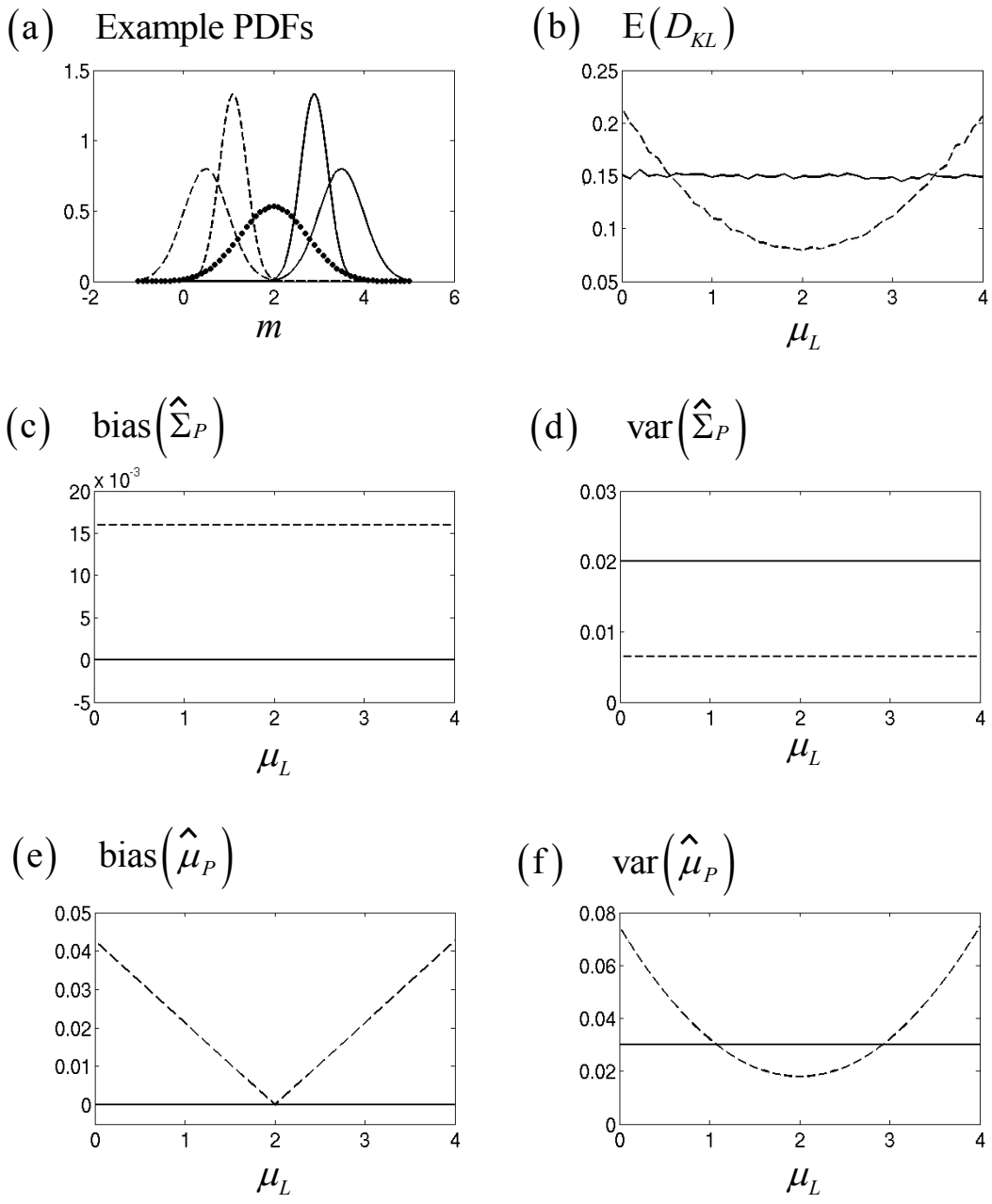

Figure F.1: Measures of the quality of the posterior estimate obtained using direct and indirect estimation were calculated for a range of posteriors, defined by: $\mu_L \in \{0, 0.1, ..., 4\}$ whilst $\mu_B = 2$, $\Sigma_B = 0.75$ and $\Sigma_L = 0.5$. (a) the old (ϕ_L) and new posterior (ϕ_P) PDF pairs for $\mu_L = 0.5$ (dashed lines) and $\mu_L = 3.5$ (solid lines). The prior PDF (ϕ_B) is plotted as a dotted bold line. (b) Average Kullback-Leibler divergences for the two methods: $\mathrm{E}\left[D_{KL}\left[p(m|d)||\hat{p}_I(m|d)\right]\right]$ and $\mathrm{E}\left[D_{KL}\left[p(m|d)||\hat{p}_D(m|d)\right]\right]$. (c) bias and (d) variance of $\hat{\Sigma}_{P_D}$ and $\hat{\Sigma}_{P_I}$. (e) bias and (f) variance of $\hat{\mu}_{P_D}$ and $\hat{\mu}_{P_I}$. In plots (b) to (f), solid lines are results obtained for the direct estimation posterior estimate (i.e., P_D), and dashed lines are for the indirect posterior estimate (i.e., P_I).

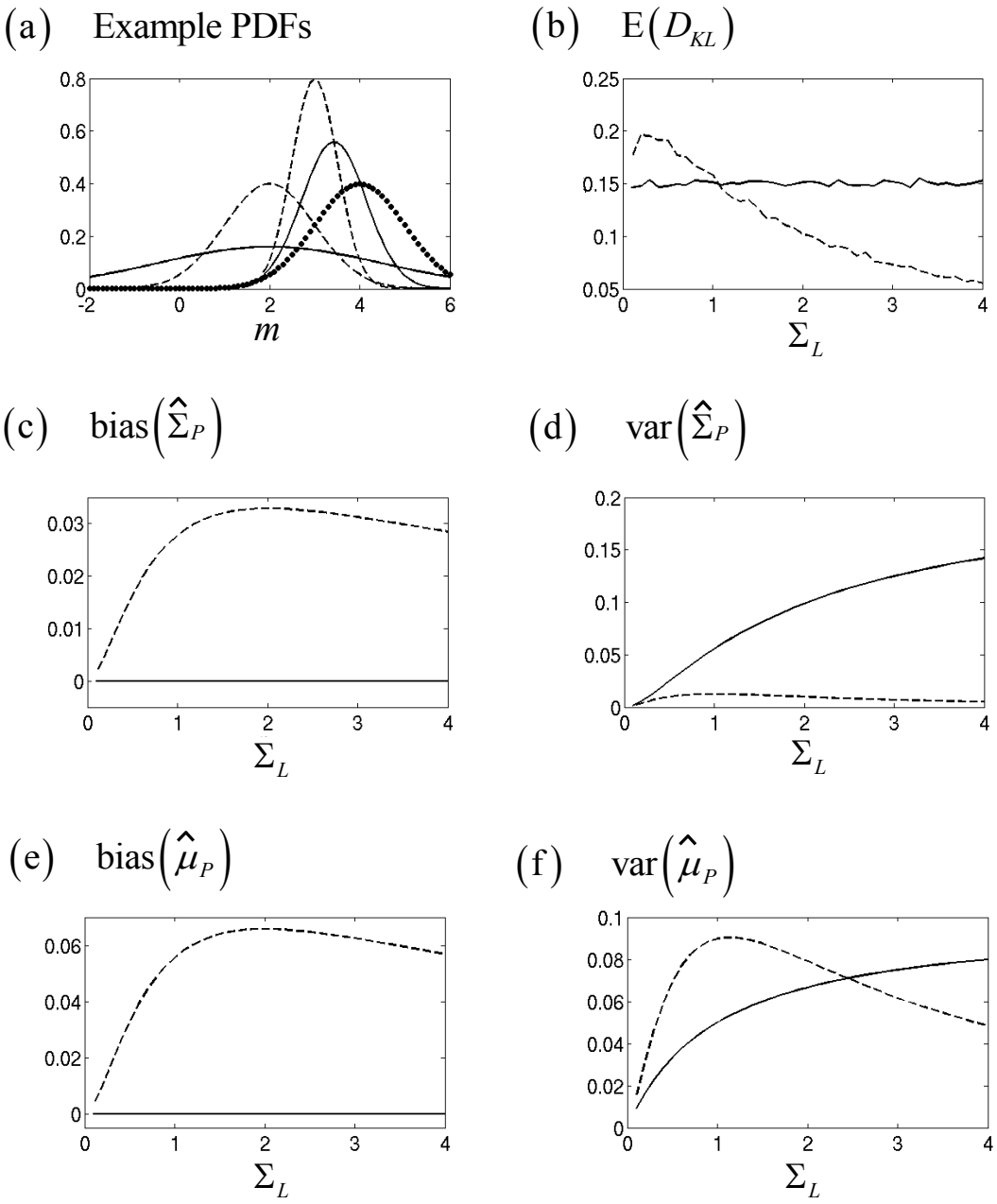

Figure F.2: Measures of the quality of the posterior estimate obtained using direct and indirect estimation were calculated for a range of posteriors, defined by: $\Sigma_L \in \{0, 0.1, ..., 4\}$ whilst $\mu_B = 4$, $\Sigma_B = 1$ and $\mu_L = 2$. (a) the old (ϕ_L) and new posterior (ϕ_P) PDF pairs for $\Sigma_L = 1$ (dashed lines) and $\Sigma_L = 2.5$ (solid lines). The prior PDF (ϕ_B) is plotted as a dotted bold line. (b) Average Kullback-Leibler divergences for the two methods: $\mathrm{E}\left[D_{KL}\left[p(m|d)||\hat{p}_I(m|d)\right]\right]$ and $\mathrm{E}\left[D_{KL}\left[p(m|d)||\hat{p}_D(m|d)\right]\right]$. (c) bias and (d) variance of $\hat{\Sigma}_{P_D}$ and $\hat{\Sigma}_{P_I}$. (e) bias and (f) variance of $\hat{\mu}_{P_D}$ and $\hat{\mu}_{P_I}$. In plots (b)-(f) solid lines are results obtained for the direct estimation posterior estimate (i.e., P_D), and dashed lines are for the indirect posterior estimate (i.e., P_I).

observed in Figures F.1(c) and F.2(c). However, a more useful observation can be made about the variance of the variance estimator, by beginning with the observation that multiplication of two Gaussians always yields a Gaussian with lower variance than either of the two Gaussians which were multiplied together (this can be seen immediately from equation F.4). This implies (given the Gaussian multiplication in equation F.3) that $\Sigma_P < \Sigma_L$. Therefore dividing equation F.23 by equation F.19 we find that

$$\frac{\text{var}\left(\hat{\Sigma}_{P_I}\right)}{\text{var}\left(\hat{\Sigma}_{P_D}\right)} = \frac{\Sigma_P^2}{\Sigma_L^2} < 1. \tag{F.26}$$

Thus to third order the variance on the variance estimator in the indirect estimation method is always less than that in the direct estimation (i.e., $\text{var}\left(\hat{\Sigma}_{P_I}\right) < \text{var}\left(\hat{\Sigma}_{P_D}\right)$). Such behaviour can be observed in Figures F.1(d) and F.2(d).

The curves corresponding to the indirect estimators in Figure F.2(c), (d), (e) and (f) all show similar features. All increase (relatively) rapidly from zero at $\Sigma_L = 0$ to reach a maximum approximately where $\Sigma_B = \Sigma_L$ and then decrease (relatively slowly) as $\Sigma_L \to \infty$. This effect can be understood, equally well for the bias and the variance, if we consider two end-member examples. The first is when the prior has infinite variance (it is 'flat') and the likelihood has zero variance (it is a delta function). No error (which would give rise to bias or variance) can be made when sampling from the old posterior (i.e., the likelihood), and we obtain a perfect posterior upon applying Bayes' rule. The second end member case is when the likelihood is flat and the prior is a delta function. In this case errors can be made when sampling the likelihood, but they are irrelevant since the prior (which we multiply by in Bayes' rule) is a delta function, and again we obtain a perfect posterior. The variance and bias of the estimators must go to zero at these end members (which correspond to either end of the horizontal axes). Between these two end members two processes compete: (i) as the likelihood variance decreases (relative to the prior variance) fewer errors occur in sampling, and (ii) as the likelihood variance increases (relative to the prior variance) these errors matter less. Thus one might expect these two competing effects to balance around the point at which the variances are equal (which is what we observe at the maxima where $\Sigma_L = 1 = \Sigma_B$).

In Figure F.1(f) we observe that $\text{var}\left(\hat{\mu}_{P_I}\right)$ tends to be lower than $\text{var}\left(\hat{\mu}_{P_D}\right)$ where the likelihood mean approaches the prior mean. This makes intuitive sense since indirect estimation makes an unbiased estimate of the mean of the likelihood. Equation

F.4 shows that as the likelihood mean approaches the prior mean, the posterior mean approaches the likelihood (and prior) mean. Thus the indirect estimate approaches a point at which it is making a direct and unbiased estimate of the posterior mean. Similarly the same mechanisms can used to explain the behaviour of bias $(\hat{\mu}_{P_I})$ in Figure F.1(e). Here we see that bias $(\hat{\mu}_{P_I})$ goes (linearly) to zero when the likelihood and prior means are equal.

As a consequence of the behaviour of the estimators described above, the overall goodness-of-fit measure, D_{KL}, tends to be lower in the indirect estimation (prior replacement) than in the direct estimation method whenever the likelihood variance is relatively large (compared to the prior variance) and/or the likelihood mean approaches the prior mean. This behaviour can be seen in Figures F.1(b) and F.2(b). Although D_{KL} is a useful, well-understood measure, it is of limited analytical use here as it cannot easily be related to the parameters of the Gaussian distributions (in the indirect estimation method). However, it neatly encapsulates the other results derived above for the biases and variances.

We found (in results not reproduced here) that varying the number of samples made, N, had little impact on the relative properties of direct and indirect estimation. N simply acts as a scaling factor in equations F.17-F.19 and F.20-F.23 (thus the variances, biases and D_{KL} all reduced with increasing N in both methods). It should also be noted that the choice of a maximum likelihood estimator here is somewhat at odds with the Bayesian framework used thus far (Ulrych et al., 2001). However we do not anticipate that attaching prior distributions to the parameters (the variance and means) could change the outcome of the analysis. For example, we could choose to use a Bayesian estimator such as the maximum a posteriori (MAP) estimator for the variance assuming a Jeffrey's prior (Lupton, 1993; Jeffreys, 1998) but we would not see any practical difference in the analytical results since this would simply change N to $N + 1$ in the expressions above (Ulrych et al., 2001).

In the next section we discuss the implications of these results for the application of prior replacement in MDN inversion (and hence geological inversion), and possible general implications for Bayesian inversion. Thus we reiterate here that direct estimation is equivalent to prior-specific training since no old posterior is used: samples are made directly from the new posterior. Also indirect estimation is equivalent to prior replacement since samples are initially made from the normalised likelihood distribution (equivalent to the old posterior with a flat old prior); then the old prior is replaced by the new prior analytically.

F.5 Discussion

It is important to note that in the results above we have assumed that we have a likelihood distribution for which we only know the unnormalised density; thus we may only estimate an old or new posterior density by first sampling from it (using MC techniques). In contrast, any prior distribution we use (whether it be the old or the new) is assumed to be known parametrically, thus we may manipulate it algebraically with respect to the estimate of the old posterior. In principle prior replacement may be performed even if the old and/or new prior is not known analytically, but the results we have obtained for this investigation of the quality of the approximation in either case would not be relevant. This is because the results assume that the Gaussian new prior is known exactly, and therefore the mean and the variance of the new prior are not random variables in our formulation. However, prior information is very often specified parametrically in geological inversion. For example, spatial correlation is often specified using Gaussian Markov random fields (Rue and Held, 2005; Eidsvik et al., 2012; Sun et al., 2012). Hence this is not a major practical limitation to the significance of these results.

If we assume that these results relating prior replacement and quality of the final posterior estimate are applicable not just for single Gaussians but for GMMs then we can explain qualitatively the results observed when prior replacement was applied to the results of MDN inversion (compared to the results of prior-specific training) in section 3.7.1. In Figure 3.3 we saw that a low probability lobe was better resolved by prior replacement than by prior-specific training. In that case the old posterior in Figure 3.1 (equivalent to the likelihood in the results above) had higher variance than the new prior in Figure 3.3(a), but had a similar mean. Thus from the estimation quality results obtained here we expect that if the old posterior variance is sufficiently large and the means sufficiently similar, that not only would the new posterior variance be more certain but also less biased in the prior replacement result (indirect estimation) than in prior-specific training (direct estimation) result. However, we would also expect that the mean would be more biased and uncertain when using prior replacement since the means of the old posterior and new prior are not identical. Thus the overall shape of the new posterior distribution should be better resolved at the expense of the exact shape of the high probability density area(s). In Figure 3.3(c) and Figure 3.3(d) this is what we observe: the peak is less well defined but the low probability lobe is much better defined when using prior

replacement.

The results of the investigation into estimation quality may have further implications. For example, suppose that we were performing such an inversion with Gaussians: can we predict a-priori whether the solution quality will be better if we do direct estimation or indirect estimation (prior replacement)? This depends on what aspect of the quality of the posterior estimate is desirable. The bias of the sample mean and variance is always lower for direct estimation than indirect estimation, but the variance of the sample variance is always less in the latter. Which method yields lower variance on the sample mean depends upon the posterior (i.e., the relative properties of the likelihood and prior), thus we cannot predict this a-priori. Similarly, we cannot predict which method will yield the smallest Kullback-Leibler divergence without calculating the posterior distribution's parameters.

Of course if we were to ask which method is better for a realistic inversion for a non-Gaussian likelihood we cannot conclude anything definitive from our results. They do support the intuitive supposition that prior replacement would yield more biased results than direct estimation. However, they also show that prior replacement can yield lower variance estimators and better overall goodness-of-fit (Kullback-Liebler divergence) for the Gaussian case. Thus for the general case it is not obvious which method to choose if these criteria are deemed to be important. In cases where the likelihood is not known analytically and we must use sampling methods, this conundrum would be useful to resolve, especially if only a limited number of samples can be made, such as in tomography problems in geophysics (Zhang et al., 2013). To our knowledge this is the first time that this issue has been raised in the literature; it should be investigated in future studies, as it may allow us to perform some Bayesian inversions more efficiently.

There is clearly similarity between prior replacement and the well-known Monte-Carlo technique of importance sampling. Importance sampling transforms samples made from a sampling (so-called 'instrumental') distribution such that they may be used to estimate the properties of another (so-called 'target') distribution. Each individual sample made from the instrumental distribution is transformed by weighting it by the ratio of its probability evaluated using the target distribution, to its probability evaluated using the instrumental distribution. Then calculation of the estimator is made using these transformed samples. As we have demonstrated for prior replacement, importance sampling can be used as a variance reduction technique. To do this the instrumental distribution should be chosen such that samples

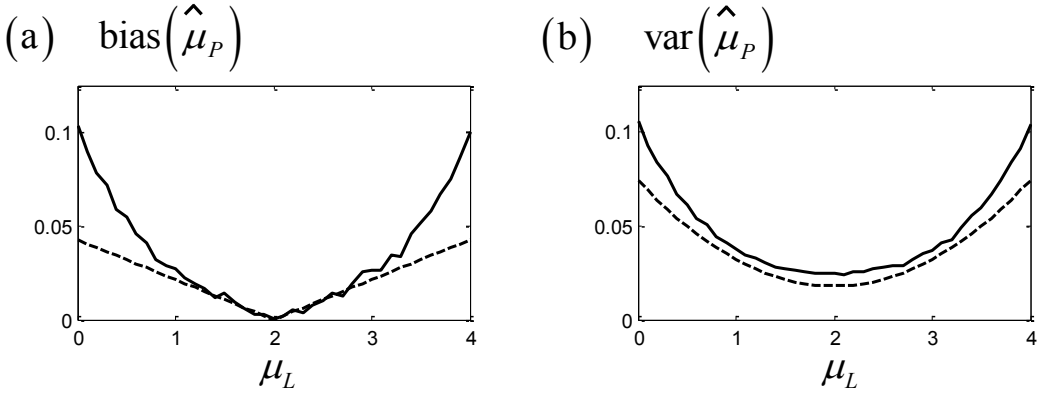

Figure F.3: An empirical comparison of prior replacement and importance sampling. The results for indirect estimation (prior replacement) have been replicated from Figure F.1 (dotted line): (a) bias and (b) variance of $\hat{\mu}_{P_I}$ for a range of posteriors, defined by $\mu_L \in \{0, 0.1, ..., 4\}$ whilst $\mu_B = 2$, $\Sigma_B = 0.75$ and $\Sigma_L = 0.5$. The equivalent results for the importance sampling estimator $\hat{\mu}_{P_{IS}}$ have been superimposed (solid line), where the old posterior (that used in indirect estimation) has been used as instrumental distribution.

are made more frequently if they are (somehow) more 'important' to the estimate required of the target distribution (compared to simply sampling directly from the target distribution). A trivial example is when attempting to estimate the mean of a target distribution. In this case if we choose an instrumental distribution which is non-zero only at the target distribution's mean value then this makes the mean estimator's variance zero (when such samples are used to estimate the mean after multiplication with the appropriate weights).

Given the above definition of importance sampling, one might expect it to yield similar, perhaps identical, results to prior replacement if the instrumental distribution is made equal to the old posterior as used in prior replacement. We now explore this hypothesis in the context of the Gaussian posterior estimation problem used above to compare direct and indirect sampling (i.e., prior replacement). To do this we first describe in more detail the importance sampling method for estimating μ_P (i.e., the mean of $p_{new}(m|d) = \phi(m; \mu_P, \Sigma_P)$).

As stated above we use as the instrumental distribution the old posterior distribution which, as defined earlier, is simply the normalised likelihood $p_{old}(m|d) = \phi(m; \mu_L, \Sigma_L)$. To estimate μ_P using importance sampling we begin by making N samples of m, $M_1, ..., M_i., ..., M_N$, from the instrumental distribution, where

$$M_i \sim p_{old}(m|d). \tag{F.27}$$

Then a weight value is calculated for each of these samples using

$$w_i = \frac{p_{new}\left(m = M_i|d\right)}{p_{old}\left(m = M_i|d\right)}. \tag{F.28}$$

Using these weights, the normalised importance sampling method (Bishop, 2006, p.533), gives the importance sampling estimator of μ_P as

$$\hat{\mu}_{P_{IS}} = \frac{1}{\sum_{i=1}^{N} w_i} \sum_{i=1}^{N} w_i M_i \tag{F.29}$$

where the IS subscript denotes the importance sampling estimate. Although the mathematical procedure of importance sampling (equations F.27 to F.29) is similar to the equivalent prior replacement operation (equations F.10 to F.14) there is a clear difference: prior replacement acts only upon the estimated mean of the old posterior to obtain the new posterior mean, whereas importance sampling acts (by applying a weighting factor) to each sample and then uses this to obtain the new posterior mean estimate. More succinctly, importance sampling transforms individual samples for later use in estimation whereas prior replacement acts to transform distributions directly.

In Figure F.3(a) and (b) we demonstrate the empirical affect of this difference between the two methods. To do this we reproduce the results obtained for the variance and bias of the prior replacement (indirect) estimate of μ_P given in Figure F.1(e) and (f) for varying values of μ_L. We compare these to the equivalent variance and bias of the $\hat{\mu}_{P_{IS}}$ estimate acquired using importance sampling using the old posterior, $p_{old}\left(m|d\right) = \phi\left(m; \mu_L, \Sigma_L\right)$, as the instrumental distribution.

There are significant differences between the variance and bias when using prior replacement and importance sampling. In general, prior replacement yields lower bias and variance. However, the overall behaviour of the bias and variance with respect to the change in μ_L is similar (i.e., the shape of the curves is similar). This suggests that we can use the same intuitive interpretation of importance sampling to understand prior replacement in terms of the importance of certain sample values in determining the required estimate, the only difference being in the way that these samples are used to obtain the final estimate. It should be noted that although this comparison has been made only for estimates of μ_P, similar results exist for the Σ_P estimators. We have omitted these for the sake of brevity since the derivation of importance sampling

for the variance estimator is not as easily exposited as that for the mean. Also we have not investigated comparison to other possible implementations of importance sampling such as the non-normalised importance sampling scheme (Bishop, 2006, p.533).

F.6 Summary

We have derived approximations for the variance and bias of estimators using prior replacement (termed indirect estimation) and compared these to sampling directly from the corresponding posterior distribution (termed direct estimation) for Gaussian prior and likelihood. Indirect estimation can outperform direct estimation when prior and likelihood have sufficiently similar means, or when the likelihood has a sufficiently large variance compared to the prior variance. Similar results were observed for the expected Kullback-Leibler divergence in each case. These results not only support our proposed use of prior replacement as a useful method for enhancing MDN training, but also highlighted possible benefits of using prior replacement rather than direct estimation in a variety of other situations where sampling is required to determine a posterior distribution. A mathematical comparison of prior replacement and the well-known Monte-Carlo technique of importance sampling was made. They were shown to be quite distinct: the former is applied to distributions, the latter to individual samples. However, empirical studies showed some similarities between results obtained with both methods suggesting that they are indeed related.

References

Bishop, C. M. (2006), *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc.

Bromiley, P. A. (2003), Products and convolutions of Gaussian distributions, *Tech. rep.*, University of Manchester.

Daniels, M. J. (1999), A prior for the variance in hierarchical models, *Canadian Journal of Statistics*, *27*(3), 567–578.

Eidsvik, J., A. O. Finley, S. Banerjee, and H. Rue (2012), Approximate Bayesian inference for large spatial datasets using predictive process models, *Computational Statistics & Data Analysis*, *56*(6), 1362–1380.

Hershey, J. R., and P. A. Olsen (2007), Approximating the Kullback Leibler divergence between Gaussian mixture models, in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007.*, pp.IV–317, IEEE.

Hobert, J. P., and G. Casella (1996), The effect of improper priors on Gibbs sampling in hierarchical linear mixed models, *Journal of the American Statistical Association*, *91*(436), 1461–1473.

Jeffreys, H. (1998), *The theory of probability*, Oxford University Press.

Kullback, S., and R. A. Leibler (1951), On information and sufficiency, *The Annals of Mathematical Statistics*, *22*(1), 79–86.

Lupton, R. (1993), *Statistics in theory and practice*, Princeton University Press.

Mosegaard, K., and M. Sambridge (2002), Monte Carlo analysis of inverse problems, *Inverse Problems*, *18*(3), R29.

Oehlert, G. W. (1992), A note on the delta method, *The American Statistician*, *46*(1), 27–29.

Rue, H., and L. Held (2005), *Gaussian Markov random fields: theory and applications*, Chapman & Hall.

Sun, D., R. K. Tsutakawa, and Z. He (2001), Propriety of posteriors with improper priors in hierarchical linear mixed models, *Statistica Sinica*, *11*(1), 77–96.

Sun, Y., B. Li, and M. G. Genton (2012), Geostatistics for large datasets, in *Advances and challenges in space-time modelling of natural events,* pp.55–77, Springer.

Tarantola, A. (2002), *Inverse problem theory: Methods for data fitting and model parameter estimation*, Elsevier Science.

Ulrych, T. J., M. D. Sacchi, and A. Woodbury (2001), A Bayes tour of inversion: A tutorial, *Geophysics*, *66*(1), 55–69.

Van der Vaart, A. W. (2000), *Asymptotic statistics*, Cambridge University Press.

Zhang, R., C. Czado, and K. Sigloch (2013), A Bayesian linear model for the high-dimensional inverse problem of seismic tomography, *The Annals of Applied Statistics*, *7*(2), 1111–1138.

Appendix G

Bias and variance of the estimators

G.1 Preliminaries

The required quantities for the indirect estimators, $\hat{\mu}_{P_I}$ and $\hat{\Sigma}_{P_I}$, are the bias of the mean, $\text{bias}\left(\hat{\mu}_{P_I}\right)$; the variance of the mean, $\text{var}\left(\hat{\mu}_{P_I}\right)$; the bias of the variance, $\text{bias}\left(\hat{\Sigma}_{P_I}\right)$; and the variance of the variance, $\text{var}\left(\hat{\Sigma}_{P_I}\right)$. $\hat{\mu}_{P_I}$ and $\hat{\Sigma}_{P_I}$ are functions of the random variables $\hat{\mu}_L$ and $\hat{\Sigma}_L$ (see equation F.14). Thus, in order to estimate the required quantities we will need to be able to approximate the expectation of a function of these random variables. Generally, if we have a function of two random variables, $f\left(\hat{\Sigma}_L, \hat{\mu}_L\right)$, then we can obtain an estimate of the expected value of that function, $\text{E}\left[f\left(\hat{\Sigma}_L, \hat{\mu}_L\right)\right]$, by using a Taylor expansion expanded around the expected value of those variables, $\boldsymbol{\theta} = \left[\text{E}\left[\hat{\Sigma}_L\right], \text{E}\left[\hat{\mu}_L\right]\right]$. The third order Taylor expansion approximation of $\text{E}\left[f\left(\hat{\Sigma}_L, \hat{\mu}_L\right)\right]$ is given by Van der Vaart (2000) as

$$\begin{aligned}\text{E}\left[f\left(\hat{\Sigma}_L, \hat{\mu}_L\right)\right] \approx & f\left(\boldsymbol{\theta}\right) + \frac{1}{2}\frac{d^2 f\left(\boldsymbol{\theta}\right)}{d\hat{\Sigma}_L^2}\text{var}\left(\hat{\Sigma}_L\right) + \frac{1}{2}\frac{d^2 f\left(\boldsymbol{\theta}\right)}{d\hat{\mu}_L^2}\text{var}\left(\hat{\mu}_L\right) \\ & + \frac{d^2 f\left(\boldsymbol{\theta}\right)}{d\hat{\Sigma}_L d\hat{\mu}_L}\text{covar}\left(\hat{\Sigma}_L, \hat{\mu}_L\right). \end{aligned} \tag{G.1}$$

Since $\hat{\mu}_L$ and $\hat{\Sigma}_L$ are estimators for a Gaussian distribution (equation F.9) we have the elementary results

$$\text{E}\left[\hat{\Sigma}_L\right] = \Sigma_L, \tag{G.2}$$

$$\text{E}\left[\hat{\mu}_L\right] = \mu_L, \tag{G.3}$$

$$\text{var}\left(\hat{\Sigma}_L\right) = \frac{2\Sigma_L^2}{N-1}, \text{ and} \tag{G.4}$$

$$\text{var}\left(\hat{\mu}_L\right) = \frac{\Sigma_L}{N}, \tag{G.5}$$

where N is the number of samples made from the old posterior. The sample mean and sample variance are independent (Riley et al., 2006, p.1230), consequently

$$\text{covar}\left(\hat{\Sigma}_L, \hat{\mu}_L\right) = 0 \tag{G.6}$$

and we may disregard this term henceforth in the Taylor expansion.

G.2 Bias of the indirect mean

The indirect mean estimator is a function of two random variables which we write as

$$\hat{\mu}_{P_I} = \left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1}\left(\Sigma_B^{-1}\mu_B + \hat{\Sigma}_L^{-1}\hat{\mu}_L\right) = f\left(\hat{\Sigma}_L, \hat{\mu}_L\right). \tag{G.7}$$

We need to estimate its expected value using the Taylor expansion such that we can estimate the bias. To do this we must first determine the derivatives. The first order derivatives of this function, f, are

$$\frac{df}{d\hat{\Sigma}_L} = \hat{\Sigma}_L^{-2}\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-2}\left(\Sigma_B^{-1}\mu_B + \hat{\Sigma}_L^{-1}\hat{\mu}_L\right) - \hat{\Sigma}_L^{-2}\mu_L\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1}, \tag{G.8}$$

and

$$\frac{df}{d\hat{\mu}_L} = \left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1}\hat{\Sigma}_L^{-1}. \tag{G.9}$$

Thus the required second order derivatives are

$$\begin{aligned}\frac{d^2f}{d\hat{\Sigma}_L^2} =& -2\hat{\Sigma}_L^{-3}\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-2}\left(\Sigma_B^{-1}\mu_B + \hat{\Sigma}_L^{-1}\hat{\mu}_L\right) \\ &+ 2\hat{\Sigma}_L^{-4}\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-3}\left(\Sigma_B^{-1}\mu_B + \hat{\Sigma}_L^{-1}\hat{\mu}_L\right) \\ &+ 2\hat{\Sigma}_L^{-3}\hat{\mu}_L\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1} - \hat{\Sigma}_L^{-4}\hat{\mu}_L\left(\Sigma_B^{-1} + \hat{\Sigma}_L^{-1}\right)^{-2}\end{aligned} \tag{G.10}$$

and

$$\frac{d^2f}{d\hat{\mu}_L^2} = 0. \tag{G.11}$$

Substituting the mean vector, $\boldsymbol{\theta} = \left[\mathrm{E}\left[\hat{\Sigma}_L\right], \mathrm{E}\left[\hat{\mu}_L\right]\right] = [\Sigma_L, \mu_L]$, and G.11 into G.1 we obtain

$$\begin{aligned}\mathrm{E}\left[\hat{\mu}_{P_I}\right] &= \mathrm{E}\left[f\left(\hat{\Sigma}_L, \hat{\mu}_L\right)\right] \\ &\approx f\left(\Sigma_L, \mu_L\right) + \frac{1}{2}\frac{d^2 f\left(\Sigma_L, \mu_L\right)}{d\hat{\Sigma}_L^2}\mathrm{var}\left(\hat{\Sigma}_L\right). \end{aligned} \tag{G.12}$$

Then substituting G.10 into this we obtain

$$\begin{aligned}E\left[\hat{\mu}_{P_I}\right] \approx & \left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-1}\left(\Sigma_B^{-1}\mu_B + \Sigma_L^{-1}\mu_L\right) \\ & + \frac{2\Sigma_L^2}{N-1}\Bigg(- \Sigma_L^{-3}\left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-2}\left(\Sigma_B^{-1}\mu_B + \Sigma_L^{-1}\mu_L\right) \\ & + \Sigma_L^{-4}\left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-3}\left(\Sigma_B^{-1}\mu_B + \Sigma_L^{-1}\mu_L\right) \\ & + \Sigma_L^{-3}\mu_L\left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-1} - \Sigma_L^{-4}\mu_L\left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-2}\Bigg). \end{aligned} \tag{G.13}$$

Noting that the posterior mean may be written

$$\mu_P = f\left(\Sigma_L, \mu_L\right) = \left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-1}\left(\Sigma_B^{-1}\mu_B + \Sigma_L^{-1}\mu_L\right), \tag{G.14}$$

and the posterior variance as

$$\Sigma_P = \left(\Sigma_B^{-1} + \Sigma_L^{-1}\right)^{-1} \tag{G.15}$$

we may then write the expected mean as

$$\begin{aligned}\mathrm{E}\left[\hat{\mu}_{P_I}\right] &\approx \mu_P + \frac{1}{N-1}\left(-\Sigma_L^{-1}\Sigma_P\mu_P + \Sigma_L^{-2}\Sigma_P^2\mu_P + \Sigma_L^{-1}\Sigma_P\mu_L - \Sigma_L^{-2}\mu_L\Sigma_P^2\right) \\ &= \mu_P + \frac{1}{N-1}\left(\mu_L - \mu_P\right)\left(\Sigma_L^{-1}\Sigma_P - \Sigma_L^{-2}\Sigma_P^2\right). \end{aligned} \tag{G.16}$$

Therefore the bias may be approximated, using its definition, as

$$\begin{aligned}\mathrm{bias}\left(\hat{\mu}_{P_I}\right) &= \mathrm{E}\left[\hat{\mu}_{P_I}\right] - \mu_P \\ &\approx \frac{1}{N-1}\left(\mu_L - \mu_P\right)\left(\Sigma_L^{-1}\Sigma_P - \Sigma_L^{-2}\Sigma_P^2\right). \end{aligned} \tag{G.17}$$

G.3 Variance of the indirect mean

When calculating the variance we wish to obtain the expected value of the squared difference between the sample mean and expected sample mean. We may write this 'residual' function g as

$$g\left(\hat{\Sigma}_L, \hat{\mu}_L\right) = \left(\hat{\mu}_{P_I} - \mathrm{E}\left[\hat{\mu}_{P_I}\right]\right)^2 = \left(f\left(\hat{\Sigma}_L, \hat{\mu}_L\right) - \mathrm{E}\left[\hat{\mu}_{P_I}\right]\right)^2. \tag{G.18}$$

The expected value of this function is the variance, that is

$$\mathrm{var}\left(\hat{\mu}_{P_I}\right) = \mathrm{E}\left[g\left(\hat{\Sigma}_L, \hat{\mu}_L\right)\right]. \tag{G.19}$$

We can use a Taylor expansion to approximate this variance. After calculating derivatives and then following a similar procedure to that in section G.2, we find an approximation for the variance of the mean estimate as

$$\mathrm{var}\left(\hat{\mu}_{P_I}\right) \approx \left(\mu_P - \mu_L\right)^2 \frac{2\Sigma_P^2 \Sigma_L^{-2}}{N-1} + \frac{\Sigma_P^2 \Sigma_L^{-1}}{N}. \tag{G.20}$$

G.4 The bias of the indirect variance

We can use a similar analysis to that in section G.2 to calculate an approximation for the bias of the variance. We begin with the function which gives the indirect posterior variance estimator,

$$\hat{\Sigma}_{P_I} = \left({\Sigma_B}^{-1} + \hat{\Sigma}_L^{-1}\right)^{-1} = h\left(\hat{\Sigma}_L\right). \tag{G.21}$$

Again, we need to estimate its expected value using the Taylor expansion such that we can estimate the bias. After doing this we find an approximation for the bias of the indirect variance as

$$\mathrm{bias}\left(\hat{\Sigma}_{P_I}\right) \approx \Sigma_P - \mathrm{E}\left[\hat{\Sigma}_{P_I}\right] = \frac{2}{N-1}\left(\Sigma_L^{-2}\Sigma_P^3 - \Sigma_L^{-1}\Sigma_P^2\right). \tag{G.22}$$

G.5 Variance of the indirect variance

When calculating the variance we wish to obtain the expected value of the squared difference between sample variance and expected sample variance. We may write

this 'residual' function as

$$r\left(\hat{\Sigma}_L\right) = \left(\hat{\Sigma}_{P_I} - \mathrm{E}\left[\hat{\Sigma}_{P_I}\right]\right)^2 = \left(h\left(\hat{\Sigma}_L\right) - \mathrm{E}\left[\hat{\Sigma}_{P_I}\right]\right)^2. \tag{G.23}$$

As previously we obtain the variance by taking the expectation

$$\mathrm{var}\left(\hat{\Sigma}_{P_I}\right) = \mathrm{E}\left[r\left(\hat{\Sigma}_L\right)\right] \tag{G.24}$$

which can be approximated using the Taylor expansion, thus we need to calculate the derivatives of r. After doing this, in a similar manner to that in section G.2, we find an approximation of the variance of the indirect sample variance of the posterior as

$$\mathrm{var}\left(\hat{\Sigma}_{P_I}\right) \approx \frac{2\Sigma_L^{-2}\Sigma_P^4}{N-1}. \tag{G.25}$$

References

Riley, K., M. Hobson, and S. Bence (2006), *Mathematical methods for physics and engineering*, Cambridge University Press.

Van der Vaart, A. W. (2000), *Asymptotic statistics*, Cambridge University Press.

Appendix H

Lists of symbols

H.1 List of symbols in Chapter 2

a_j^l	neural network (NN) node variable in layer l
$\mathbf{a}^l$	vector containing all node variables in layer l of NN
b	index to traces of synthetic data
B	number of traces of synthetic data generated
$\mathbf{d}$	AVA-type data for 1-D grid/trace
$\mathbf{d}_\mathbf{x}^r$	real AVA-type data down a trace at lateral position $\mathbf{x}$ in Laggan dataset
$\mathbf{d}_b^s$	synthetic AVA-type data trace generated from $\mathbf{e}_b^s$ for Laggan dataset
$\mathbf{e_0}$	mean vector (or initial model) for low-fidelity Gaussian prior
E_N	sum-of-squares error for NN training/validation dataset
η	learning rate parameter for back-propagation
E[ ]	expectation operator
$\mathbf{e}$	true elastic parameters down 1-D grid/trace
$\hat{\mathbf{e}}$	elastic parameter (deterministic) estimates down 1-D grid/trace
$\mathbf{e}_\mathbf{x}^r$	true (real) elastic parameters down a trace at lateral position $\mathbf{x}$ in Laggan dataset
$\hat{\mathbf{e}}_\mathbf{x}^r$	real deterministic elastic parameter estimates data down a trace at lateral position $\mathbf{x}$ in Laggan dataset
$\mathbf{e}_b^s$	the b^{th} trace of synthetic elastic parameters sampled from $p_H(\mathbf{e})$ for Laggan dataset

$\hat{\mathbf{e}}_b^s$	synthetic deterministic elastic parameter estimates obtained by inverting $\mathbf{d}_b^s$ for Laggan dataset
$g(x)$	sigmoidal activation function in NN
i	general index (no fixed definition)
j	general index (no fixed definition)
k	general index (no fixed definition)
K^l	number of nodes in layer l (not including the bias node) of NN
L	number of layers in NN (not including input layer)
λ	approximation length for q
N	number of training instances in NN training dataset
ϕ	masking noise percentage for pre-training
$p_H(\mathbf{e})$	high-fidelity prior distribution
$p_L(\mathbf{e})$	low-fidelity prior distribution
q	NN function
$\mathcal{Q}$	Operation which extracts $[\mathbf{u}^s, \mathbf{v}^s]$ pairs from $[\hat{\mathbf{e}}^s, \mathbf{e}^s]$
$\mathbf{R}$	reflectivity vector
^{r}	superscript, denotes quantities derived from real data
^{s}	superscript, denotes quantities derived from synthetic data
$\mathbf{S}$	wavelet block matrix specifying $[\mathbf{w}_{near}, \mathbf{w}_{mid}, \mathbf{w}_{far}]$
$\mathbf{\Sigma_d}$	covariance matrix for AVA-type data
$\mathbf{\Sigma_e}$	covariance matrix for low-fidelity Gaussian prior
$\mathbf{u}$	input vector for q
$\mathbf{v}$	vector defining output of q (i.e., E[$\mathbf{v}$])
w_{ij}^l	NN weight in layer l
$\mathbf{W}$	matrix containing all NN weights
$\mathbf{x}$	used to denote lateral position $[x, y]$ in Laggan dataset
z	vertical coordinate of cell

H.2 List of symbols in Chapter 3

α_j	weight of j^{th} kernel in Gaussian mixture model (GMM)
$\mathbf{e}_i$	elastic parameter vector in cell i, where $\mathbf{e}_i = [I_P, I_S]_i$
i	index to cell in grid
j	index to kernel in GMM
k	normalising constant
K	number of kernels in GMM
M	total number of cells in 2-D model grid
$\mathbf{m}_i$	continuous geological parameters in cell i
m_1	clay content by volume parameter used in example application
m_2	sandstone matrix porosity parameter used in example application
$\boldsymbol{\mu}_j$	mean of j^{th} kernel in GMM
$p(\mathbf{e}_i\|\mathbf{m}_i)$	cell-wise geological likelihood (for $\mathbf{m}_i$)
p_{new}	refers to distribution (prior or posterior) for 'new' situation
p_{old}	refers to distribution (prior or posterior) for 'old' situation
$\phi(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
q	total number of times the prior changes (equal to M in the 2-D reservoir grid example)
$\boldsymbol{\Sigma}_j$	covariance matrix of j^{th} kernel in GMM

H.3 List of symbols in Chapter 4

a	approximation length parameter
α	probability of transition in Metropolis-Hastings (MH) algorithm
b	proposed approximation length parameter in 3-D
C	number of cliques on the grid
$\mathbf{e}$	set of all $\mathbf{e}_i$ in the grid $\mathbf{e} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_M]$
$\mathbf{e}_i$	elastic parameter vector in cell i, where $\mathbf{e}_i = [I_P, I_S]_i$
$f_j(\mathbf{g}_{\Lambda_j})$	function over a clique
g_i	the discrete geological parameter in cell i
$\mathbf{g}'$	candidate sample in MH algorithm
$\mathcal{G}'$	proposed reduced sample size for $\mathbf{g}$
$\mathcal{G}$	sample space of g_i
$\mathbf{g}$	set of all g_i in the grid $\mathbf{g} = [g_1, g_2, ..., g_M]$
$\mathcal{G}^M$	sample space of $\mathbf{g}$
$\mathcal{H}$	the set of all indices in the grid $\mathcal{H} = [1, 2, ..., M]$
i	index to cell in grid (used in recursive algorithm)
j	index used in recursive algorithm
k	maximum index (number) in $Ne(i)$
$l(z)$	operator used to select a set of rows around z to form a sub-grid
Λ	a clique (set)
M	the total number of indices in the grid
n	number of samples made in MH algorithm
$Ne(i)$	neighbourhood of cell i
q	proposal distribution in MH algorithm
S	length (cells) of a square neighbourhood's sides
t	iteration of Gibbs sampling algorithm
$\mathcal{U}[\mathcal{L}]$	Uniform distribution, non-zero only over $\mathcal{L}$

H.4 List of symbols in Chapter 5

α	magnitude of perturbation to an element of $\mathbf{T}$
α_1	the α parameter used in the first stage of the algorithm
α_2	the α parameter used in the second stage of the algorithm
β	probability that an element of $\mathbf{T}$ will be perturbed
β_1	the β parameter used in the first stage of the algorithm
β_2	the β parameter used in the second stage of the algorithm
$\mathcal{C}$	the set of all possible configurations of $Ne(i)$
$\lvert\mathcal{C}\rvert$	number of possible configurations of $Ne(i)$, and hence statistics (probabilities) in $\mathbf{T}$
g_i	the discrete geological parameter in cell i
$\mathbf{g}$	set of all g_i in the grid $\mathbf{g} = [g_1, g_2, ..., g_M]$
$\mathbf{g}_j$	the j^{th} realisation of $\mathbf{g}$ (in $\mathcal{R}$) simulated using $\mathbf{T}_j$
$\mathbf{g}_{best}$	realisation made using ideal statistics vector
$\mathbf{g}_{rank=1}$	the number 1 ranked realisation in $\mathcal{R}$
$\mathbf{g}_{target}$	the target pore-space realisation displayed to the expert
i	index of a cell (geological parameter) in the grid
j	index to members of population, $j \in [1, ..., P]$
k	index used to reference statistics in $\mathbf{T}$
l	iteration number of elicitation algorithm
M	the total number of indices in the grid
$Ne(i)$	neighbourhood of cell i
P	the number of individuals in a population
P^*	number of population members to be ranked
$\mathcal{R}$	set of P realisations corresponding to $\mathcal{S}$
$\mathcal{S}$	population (set) of P $\mathbf{T}$ vectors
t_k	the k^{th} statistic in the statistics vector in $\mathbf{T}$
$\mathbf{T}$	statistics vector $\mathbf{T} = \{t_k \mid k \in \{1, 2, ..., L\}\}$
$\mathbf{T}_j$	the j^{th} $\mathbf{T}$ vector in $\mathcal{S}$
$\mathbf{T}_{best}$	ideal statistics vector
$\mathbf{T}_{rank=1}$	the $\mathbf{T}$ vector corresponding to $\mathbf{g}_{rank=1}$
$\mathbf{T}_{target}$	the $\mathbf{T}$ vector used to generate $\mathbf{g}_{target}$
$\mathcal{U}[\mathcal{L}]$	Uniform distribution, non-zero only over $\mathcal{L}$