GEOPHYSICS, VOL. 86, NO. 6 (NOVEMBER-DECEMBER 2021); P. F61–F69, 6 FIGS.
10.1190/GEO2021-0055.1

# SeReMpy: Seismic reservoir modeling Python library

Dario Grana[1] and Leandro de Figueiredo[2]

## ABSTRACT

Seismic reservoir characterization is a subfield of geophysics that combines seismic and rock-physics modeling with mathematical inverse theory to predict reservoir variables from the measured seismic data. An open-source comprehensive modeling library that includes the main concepts and tools is still missing. We have developed a Python library called SeReMpy with state-of-the-art of seismic reservoir modeling for reservoir properties characterization using seismic and rock-physics models and Bayesian inverse theory. The most innovative component of the library is the Bayesian seismic and rock-physics inversion to predict the spatial distribution of petrophysical and elastic properties from seismic data. The inversion algorithms include Bayesian analytical solutions of the linear-Gaussian inverse problem and Markov chain Monte Carlo (MCMC) numerical methods for nonlinear problems. The library includes four modules: geostatistics, rock physics, facies, and inversion, as well as several scripts with illustrative examples and applications. We illustrate the use of the functions of the module and develop codes for practical inversion problems using synthetic and real data. The applications include a rock-physics model for the prediction of elastic properties and facies using well-log data, a geostatistical simulation of continuous and discrete properties using well logs, a geostatistical interpolation and simulation of 2D maps of temperature, an elastic inversion of partial stacked seismograms with Bayesian linearized amplitude-variation-with-offset inversion, a rock-physics inversion of partial stacked seismograms with MCMC methods, and a 2D seismic inversion.

## INTRODUCTION

Seismic reservoir characterization aims to improve the description of the reservoir in terms of petrophysical and elastic variables using seismic data. In general, seismic reservoir characterization requires the integration of multiple fields including seismology for modeling the seismic response, rock physics for modeling the elastic response, petrophysics for modeling the petrophysical response, geostatistics for stochastic simulating random variables with spatial correlation, inverse theory for predicting the model variables from the geophysical response, and Bayesian statistics for quantifying the uncertainty in the model variables. A detailed review of seismic reservoir characterization methods is given in Doyen (2007) and Grana et al. (2021).

In this paper, we focus on Bayesian methods for geophysical inverse problems to estimate the spatial distribution of a set of model variables from the measured data, predict the most likely values, and quantify the uncertainty of the properties of interest. In the Bayesian setting, the solution of the inverse problem is the posterior probability density function (PDF) of the model variables conditioned on the measured data (Scales and Tenorio, 2001; Ulrych et al., 2001; Tarantola, 2005). The application of Bayesian inversion to seismic reservoir characterization problems gained increasing attention in the past few decades. The precursive works on probabilistic and stochastic approaches to seismic inverse problems date back to the 1980s and include publications by Tarantola and Valette (1982), Doyen (1988), Bortoli et al. (1993), Haas and Dubrule (1994), Mosegaard and Tarantola (1995), and Sen and Stoffa (1996). The progress on seismic inversion proceeded parallel to advances in geostatistics for reservoir modeling (Journel and Huijbregts, 1978; Deutsch and Journel, 1998; Deutsch, 2002).

Subsequent publications focus on stochastic inversion methods to predict rock and fluid properties from seismic data (Bosch, 1999; Mukerji et al., 2001; Buland and Omre, 2003; Eidsvik et al., 2004; Gun-

Manuscript received by the Editor 28 January 2021; revised manuscript received 25 April 2021; published ahead of production 19 July 2021; published online 24 September 2021.
[1]University of Wyoming, Department of Geology and Geophysics, School of Energy Resources, Laramie, Wyoming 82071, USA. E-mail: dgrana@uwyo.edu (corresponding author).
[2]LTrace, Florianópolis, Santa Catarina 88032-005, Brazil. E-mail: leandrop.fgr@gmail.com.

ning and Glinsky, 2004; Coléou et al., 2005; Contreras et al., 2005). Reviews of these methods are given in Doyen (2007) and Bosch et al. (2010). Buland and Omre (2003) present an elegant and efficient analytical solution of the seismic amplitude-variation-with-offset (AVO) inversion problem using Bayesian linear-Gaussian inverse theory. Subsequent publications combining seismic and rock-physics modeling with Bayesian inversion rely on the Buland and Omre formulation and extend the approach to petrophysical inversion and lithofluid facies classification (Larsen et al., 2006; Buland et al., 2008; Grana and Della Rossa, 2010; Ulvmoen and Omre, 2010; Rimstad et al., 2012; Grana, 2016, 2018; Jullum and Kolbjørnsen, 2016; Grana et al., 2017; de Figueiredo et al., 2018; Fjeldstad and Grana, 2018). Stochastic methods with similar probabilistic formulations have been proposed by Connolly and Kemper (2007), González et al. (2008), Gunning and Glinsky (2007), Spikes et al. (2008), Azevedo et al. (2013), and Kemper and Gunning (2014). Alternatively, stochastic methods for seismic reservoir characterization based on Monte Carlo or Markov chain Monte Carlo (MCMC) algorithms to sample from the posterior distribution of the variables of interest have been presented in Mosegaard and Tarantola (1995), Sambridge and Mosegaard (2002), Eidsvik et al. (2004), Connolly and Hughes (2016), Jeong et al. (2017), and de Figueiredo et al. (2019a, 2019b).

All of the previously mentioned publications include formulations with various model parameterizations, algorithms, and assumptions, but they all have in common the use of probabilistic methods to improve the reservoir characterization from seismic data and well logs. Some of the available publications include open-source codes (Gunning and Glinsky, 2004; Hansen, 2004; Pebesma, 2004; Liu and Grana, 2019). Dedicated toolboxes are also available for specific tasks, such as rock-physics modeling (Avseth et al., 2010) and geostatistics (Deutsch and Journel, 1998; Remy et al., 2009). However, a comprehensive library for seismic reservoir characterization including seismic and rock-physics models, geostatistical algorithms, and inversion methods is still missing. We present here the open-source library SeReMpy, a Python library for seismic reservoir modeling, available in the dedicated GitHub repository (see the "Data and materials availability" section).

The SeReMpy library includes four modules for facies, geostatistics, inversion, and rock-physics modeling as well as several examples. The facies module includes methods for facies classification (Doyen, 2007), the geostatistics module includes algorithms for interpolation and simulation of spatially correlated random variables (Deutsch, 2002), the rock-physics module includes rock-physics models for elastic properties (Mavko et al., 2020), and the inversion module includes the state-of-the-art of Bayesian methods for seismic reservoir characterization. In this paper, we focus on the inversion methods. First, we review the theory of Bayesian inversion for seismic and petrophysical characterization, we then present all the functionalities of the library, and we finally show several applications to seismic reservoir characterization problems.

## THEORY

One of the goals of seismic reservoir characterization is to predict a set of unknown geophysical variables of interest $\mathbf{m}$ from a collection of measurements $\mathbf{d}$. Measured data typically include poststack or partial-stacked seismic data, whereas the geophysical variables might include elastic properties (P- and S-wave velocity and density), petrophysical properties (porosity, mineral volumes, and fluid saturations), or categorical variables such as facies or lithologies.

Geophysical models are mathematical operators $\mathbf{f}:\mathbb{R}^{n_m} \to \mathbb{R}^{n_d}$ that approximate the relation between the model variables $\mathbf{m}$ and the data $\mathbf{d}$:

$$\mathbf{d} = \mathbf{f}(\mathbf{m}) + \boldsymbol{\varepsilon}, \tag{1}$$

where the term $\boldsymbol{\varepsilon}$ represents the error associated with the measurements, $n_m$ is the number of unknown variables, and $n_d$ is the number of measurements.

The forward operator $\mathbf{f}$ depends on the parameterization of the problem and the available data, but it generally includes a seismic model to compute the seismic response and/or a rock-physics model to calculate the elastic response of a vertical sequence of rocks with known petrophysical properties. The forward operator is generally discretized due to the finite number of measurements, and it is often simplified using linear approximations. In this case, given the measurements $\mathbf{d} = \{d_i\}$ $(i = 1, \ldots, n_d)$, $\mathbf{f}$ is written as a matrix $\mathbf{F}$ of dimensions $n_d \times n_m$ and equation 1 becomes a linear system of equations:

$$\mathbf{d} = \mathbf{F}\mathbf{m} + \boldsymbol{\varepsilon}, \tag{2}$$

where $\mathbf{m}$ is a vector of length $n_m$ and $\boldsymbol{\varepsilon}$ is a vector of length $n_d$.

From a Bayesian perspective, the solution of the inverse problem is the posterior PDF of the model $\mathbf{m}$, $P(\mathbf{m}|\mathbf{d})$, and it can be computed using the Bayes' theorem:

$$P(\mathbf{m}|\mathbf{d}) = \frac{P(\mathbf{d}|\mathbf{m})P(\mathbf{m})}{P(\mathbf{d})} = \frac{1}{c}P(\mathbf{d}|\mathbf{m})P(\mathbf{m}), \tag{3}$$

where $P(\mathbf{d}|\mathbf{m})$ is the likelihood function, $P(\mathbf{m})$ is the prior distribution, and $P(\mathbf{d})$ is a normalizing constant ($P(\mathbf{d}) = c$).

In a statistical setting, calculating the solution of equation 1 is a maximum likelihood estimation problem. If we assume that the observations are independent and $f_i(d_i|\mathbf{m})$ represents the PDF of the measurement $d_i$ $(i = 1, \ldots, n_d)$ with Gaussian noise $\varepsilon_i \sim \mathcal{N}(\varepsilon_i; 0, \sigma_i^2)$, then the joint probability density of the vector of independent observations $\mathbf{d}$ is

$$f(\mathbf{d}|\mathbf{m}) = \prod_{i=1}^{n_d} f_i(d_i|\mathbf{m}) = \frac{1}{(2\pi)^{n_d/2}\prod_{i=1}^{n_d}\sigma_i}\prod_{i=1}^{n_d} e^{-\frac{(d_i-(\mathbf{Fm})_i)^2}{2\sigma_i^2}}. \tag{4}$$

In the maximum likelihood estimation, we aim to estimate the model $\mathbf{m}$ that maximizes the likelihood function in equation 4 (Tarantola, 2005).

In the linear-Gaussian case, the solution of the inverse problem can be estimated analytically (Buland and Omre, 2003; Tarantola, 2005). We consider a linear operator $\mathbf{f}$ with matrix $\mathbf{F}$. We assume that the prior distribution of the model vector $\mathbf{m}$ is Gaussian $\mathcal{N}(\mathbf{m}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ with prior mean $\boldsymbol{\mu}_m$ and spatially correlated prior covariance matrix $\boldsymbol{\Sigma}_m = \boldsymbol{\Sigma}_m^0 \otimes \boldsymbol{\Sigma}_\tau$ given by the Kronecker product of the stationary covariance of the model variables $\boldsymbol{\Sigma}_m^0$ and the spatial correlation matrix $\boldsymbol{\Sigma}_\tau$ defined by a prior spatial correlation function $\upsilon(\tau)$. We also assume that the measurement error vector $\boldsymbol{\varepsilon}$ is Gaussian $\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{\varepsilon}; \mathbf{0}, \boldsymbol{\Sigma}_\varepsilon)$ with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}_\varepsilon$. Then, the posterior distribution $P(\mathbf{m}|\mathbf{d})$ is Gaussian $\mathcal{N}(\mathbf{m}; \boldsymbol{\mu}_{m|d}, \boldsymbol{\Sigma}_{m|d})$ with conditional mean $\boldsymbol{\mu}_{m|d}$:

$$\boldsymbol{\mu}_{m|d} = \boldsymbol{\mu}_m + \boldsymbol{\Sigma}_m\mathbf{F}^T(\mathbf{F}\boldsymbol{\Sigma}_m\mathbf{F}^T + \boldsymbol{\Sigma}_\varepsilon)^{-1}(\mathbf{d} - \mathbf{F}\boldsymbol{\mu}_m) \tag{5}$$

and conditional covariance matrix $\mathbf{\Sigma}_{m|d}$:

$$\mathbf{\Sigma}_{m|d} = \mathbf{\Sigma}_m - \mathbf{\Sigma}_m \mathbf{F}^T (\mathbf{F}\mathbf{\Sigma}_m \mathbf{F}^T + \mathbf{\Sigma}_\varepsilon)^{-1} \mathbf{F}\mathbf{\Sigma}_m. \tag{6}$$

Buland and Omre (2003) present a Bayesian linearized AVO inversion in which the formulation in equations 5 and 6 is applied to the seismic inversion problem. The model variables are the elastic properties (P- and S-wave velocity and density), and the linear forward operator is the convolution of a known wavelet and the linearized AVO approximation of the reflectivity coefficients (Buland and Omre, 2003). The matrix $\mathbf{F}$ is expressed as a matrix multiplication, $\mathbf{F} = \mathbf{WAD}$, where $\mathbf{W}$ is the wavelet Toeplitz matrix, $\mathbf{A}$ is the reflectivity coefficients matrix, and $\mathbf{D}$ is a first-order differential matrix and it is applied to the logarithm of the elastic properties, $\mathbf{d} = \mathbf{WAD}\ \ln(\mathbf{m}) + \mathbf{e}$. If a linear rock-physics model is available to relate the reservoir properties $\mathbf{r}$ to the logarithm of the elastic properties, $\ln(\mathbf{m})$, then the Bayesian inversion can be extended to the joint seismic and rock-physics problem $\mathbf{d} = \mathbf{WADRr} + \mathbf{e}$ as shown in Grana et al. (2017) and Fjeldstad and Grana (2018).

For nonlinear problems, numerical methods such as MCMC algorithms are adopted. MCMC algorithms are iterative methods in which at each iteration, a model realization $\mathbf{m}_{i+1}$ is sampled from a proposal distribution conditioned on the model $\mathbf{m}_i$ at the previous iteration and it is accepted or rejected according to their probability ratio. The chain of accepted models asymptotically converges to the posterior distribution of the model properties conditioned on the data and represents the solution of the inverse problem. The most common MCMC method for seismic inversion problems is the Metropolis algorithm, in which the proposed model realization $\mathbf{m}'$ is drawn from a symmetric proposal distribution $g(\mathbf{m}'|\mathbf{m}_i) = g(\mathbf{m}_i|\mathbf{m}')$ and accepted with probability $p$

$$\begin{aligned} p &= \min\left\{ \frac{P(\mathbf{d}|\mathbf{m}')P(\mathbf{m}')g(\mathbf{m}_i|\mathbf{m}')}{P(\mathbf{d}|\mathbf{m}_i)P(\mathbf{m}_i)g(\mathbf{m}'|\mathbf{m}_i)}, 1 \right\} \\ &= \min\left\{ \frac{P(\mathbf{d}|\mathbf{m}')P(\mathbf{m}')}{P(\mathbf{d}|\mathbf{m}_i)P(\mathbf{m}_i)}, 1 \right\}. \end{aligned} \tag{7}$$

At each iteration, we randomly generate a random number $u$ according to a uniform distribution $U([0,1])$. If $u < p$, $\mathbf{m}'$ is accepted (i.e., $\mathbf{m}_{i+1} = \mathbf{m}'$); otherwise, $\mathbf{m}'$ is rejected (i.e., $\mathbf{m}_{i+1} = \mathbf{m}_i$). The Metropolis algorithm is a special case of the more general Metropolis-Hasting algorithms in which the proposal distribution is assumed to be symmetric rather than a general PDF.

We assume that the prior distribution $P(\mathbf{m})$ is Gaussian $\mathcal{N}(\mathbf{m}; \boldsymbol{\mu}_m, \mathbf{\Sigma}_m)$ with prior mean $\boldsymbol{\mu}_m$ and spatially correlated prior covariance matrix $\mathbf{\Sigma}_m = \mathbf{\Sigma}_m^0 \otimes \mathbf{\Sigma}_\tau$, and that the likelihood function $P(\mathbf{d}|\mathbf{m})$ is Gaussian $\mathcal{N}(\mathbf{d}; \mathbf{f}(\boldsymbol{\mu}_m), \mathbf{\Sigma}_\varepsilon)$, and then the acceptance probability $p$ can be written as

$$\begin{aligned} p = \min\Bigg\{ &\exp\left( -\frac{1}{2}((\mathbf{d}' - \mathbf{d})^T \mathbf{\Sigma}_d^{-1} (\mathbf{d}' - \mathbf{d}) \right. \\ &\left. - (\mathbf{d}_i - \mathbf{d})^T \mathbf{\Sigma}_d^{-1} (\mathbf{d}_i - \mathbf{d})) \right) \\ \times &\exp\left( -\frac{1}{2}((\mathbf{m}' - \boldsymbol{\mu}_m)^T \mathbf{\Sigma}_m^{-1} (\mathbf{m}' - \boldsymbol{\mu}_m) \right. \\ &\left. - (\mathbf{m}_i - \boldsymbol{\mu}_m)^T \mathbf{\Sigma}_m^{-1} (\mathbf{m}_i - \boldsymbol{\mu}_m)) \right), 1 \Bigg\}, \end{aligned} \tag{8}$$

where $\mathbf{d}' = \mathbf{f}(\mathbf{m}')$, $\mathbf{\Sigma}_d = \mathbf{\Sigma}_\varepsilon \otimes \mathbf{I}_{n_d}$, and $\mathbf{d}_i = \mathbf{f}(\mathbf{m}_i)$. After convergence, the statistical estimators of the posterior distributions are calculated from a set of model realizations, excluding the realizations in the "burn-in" period in which the chain still depends on the initial model realization.

An efficient compromise between linearized analytical solutions and MCMC iterative solutions is given by ensemble-based methods, such as the ensemble Kalman filter (EnKF), ensemble smoother (ES), and ES with multiple data assimilation (ES-MDA) and have been used in reservoir engineering models (Emerick and Reynolds, 2013) and geophysical studies (Liu and Grana, 2018). These methods are based on a Bayesian updating step of an ensemble of prior models. The EnKF assimilates data sequentially in time, the ES assimilates all data simultaneously in a single update step, and the ES-MDA iteratively assimilates the same data multiple times. In general, ensemble methods are more efficient than MCMC but the limited ensemble size might affect the uncertainty quantification.

## CODE DESCRIPTION

The SeReMpy library includes four modules and a data folder that contains multiple synthetic data sets used for the examples and the elevation-temperature data set from Yellowstone National Park. The folder Output contains sample output files of the proposed scripts. In this section, we describe the four modules and introduce the scripts with illustrative examples for their use. Additional information on the external libraries used in SeReMpy and the functions included in the modules are available in the documentation file included in the code package.

The *RockPhysics.py* module contains functions with several rock-physics models. For a reference of the derivation of the rock-physics models used in this module, we refer to Avseth et al. (2010), Dvorkin et al. (2014), and Mavko et al. (2020). The goal of this module is to provide a set of functions that calculate the elastic response in terms of P- and S-wave velocity and density of a saturated porous rock with known porosity, mineral volumes, and fluid saturations. The density is computed using the function *DensityModel*, whereas the P- and S-wave velocities are computed using one of the following models: Han's linearized model, Wyllie's equation, Raymer's equation, Dvorkin's soft sand and stiff sand models, the spherical inclusion model, and Berryman's inclusion models (Mavko et al., 2020). For porous rocks with multiple mineral components and/or multiple fluid components, the elastic properties (elastic moduli and density) of the effective solid and fluid phases are computed using the function *MatrixFluidModel*.

The *Geostats.py* module contains functions for kriging and geostatistical simulations of discrete and continuous random variables. For the derivation of the geostatistical algorithms, we refer readers to Deutsch (2002) and Deutsch and Journel (1998) for the FORTRAN library GSLIB. The *Geostats.py* module contains functions for simple kriging and ordinary kriging for the interpolation of continuous properties and for indicator kriging for the interpolation of discrete properties given a set of sparse measurements and their locations and a selected spatial correlation model (Deutsch, 2002). It also includes functions for sequential Gaussian simulation (SGS) for sampling continuous properties and for sequential indicator simulation for sampling discrete properties with spatial correlation and conditioned on the available measurements (Deutsch, 2002). These functions are based on the concept of spatial covariance functions (or variograms). Isotropic and anisotropic spatial covariance functions can be generated for 1D and 2D applications using the functions *SpatialCovariance1D* and *SpatialCovariance2D* and one of the available theoretical functions (exponential, Gaussian, and spherical models). Anisotropic kriging interpolation and simulations require the use of the function *SpatialCovariance2D*. Correlated simulations of multiple random variables can also be sequentially generated using a local variable mean computed according to a linear regression model. For efficient implementations of the simulation methods for 2D and 3D problems, we recommend the open-source software SGeMS (Remy et al., 2009), the GSLIB library (Deutsch and Journel, 1998), and the MATLAB toolbox mGstat (Hansen, 2004).

The *Inversion.py* module contains functions for seismic and rock-physics inversion using analytical and numerical solutions. The available functions include multiple methods for seismic and petrophysical inversion. The function *SeismicInversion* implements the Bayesian linearized AVO seismic inversion based on the approach presented in Buland and Omre (2003). The linearized forward model is available in the function *SeismicModel*. Five alternative algorithms are proposed for rock-physics inversion according to various statistical assumptions: a linear Gaussian model, a linear Gaussian mixture model, a nonlinear Gaussian model, a nonlinear Gaussian mixture model, and a nonlinear nonparametric model, according to the methods presented in Grana and Della Rossa (2010) and Grana (2016, 2018). Inversions based on stochastic sampling and optimization algorithms, such as ES-MDA (Liu and Grana, 2018) and MCMC (de Figueiredo et al., 2018) are also implemented. The module also provides functions for data transformation and back transformation of non-Gaussian distributions.

The *Facies.py* module contains functions for facies classification from multiple attributes, including the Gaussian and nonparametric methods proposed in Doyen (2007).

The SeReMpy library also includes multiple scripts with several examples. In the following, we describe the tutorial scripts to illustrate how to use the modules. The script *RockPhysicsModelDriver.py* shows the application of rock-physics models to calculate the P- and S-wave velocity and density from a vector of porosity values of a well log. The input data are provided in the ASCII file data1.dat. This example assumes a single mineral (quartz) and a single fluid component (water). For application to data sets with multiple mineral and fluid components, the user should compute the effective bulk and shear moduli and density of the solid and fluid phases given the volumetric fractions. An example of such a model is given in the "Applications" section. The script also shows various plotting tools to visualize the vector of model predictions as a well log or in the petroelastic domain.

The script *GeostatsContinuousDriver.py* shows an illustrative example for geostatistical methods for interpolation and simulation of a continuous random variable, based on a set of available measurements and a spatial correlation model. In the first example, in which four measurements of P-wave velocity are available, we interpolate and simulate the value of the velocity at a given location, using an exponential covariance function. The interpolation is obtained using simple or ordinary kriging, whereas the simulation is obtained using SGS. In the second example, we apply the geostatistical methods to the elevation data set of a region of Yellowstone National Park. For this data set, the true elevation map is available. From the true map, we extract 15 measurements, and we interpolate and simulate the elevation values at all of the remaining locations. The script *GeostatsDiscreteDriver.py* replicates the same examples but using a discrete random variable as the property of interest. In the first example with P-wave velocity measurements, the discrete property represents lithologic facies (sand and shale), whereas in the elevation example, the discrete property represents topographic features (valleys and peaks). The script *GeostatsDiscreteDriver.py* also includes a simulation of a discrete random variable using Markov chain models that can be used as a simulation method to generate vertical sequences of facies or lithologies.

The script *SeismicModelDriver.py* shows the seismic forward model based on the Buland and Omre approach. The forward operator is a convolution of a wavelet and the Aki-Richards' linearized approximation of the Zoeppritz equations. The script applies the function to the data set in the ASCII file data2.dat that includes the well logs of the P- and S-wave velocity and density and computes three partial-stacked seismograms according to three incident angles. The script *SeismicInversionDriver.py* implements the Bayesian linearized AVO inversion based on the Buland and Omre approach. The script applies the function to the data set in the ASCII file data3seis.dat that includes three partial-stacked seismograms according to three incident angles and computes the corresponding P- and S-wave velocity and density. The script *RockPhysicsInversionDriver.py* shows the Bayesian rock-physics inversion methods, for linear and nonlinear forward models with various statistical assumptions for the PDF of the model variables (Gaussian, Gaussian mixture, and nonparametric). The script applies the function to the data set in the ASCII file data4.dat that includes the well logs of the P- and S-wave velocity and density and computes the corresponding porosity, clay volume, and water saturation. The scripts *ESSeisInversionDriver.py* and *ESPetroInversionDrive.py* show the ES-MDA inversion method for seismic and petrophysical inversion, respectively. The seismic inversion is applied to the data in the ASCII file data3seis.dat, whereas the petrophysical inversion is applied to the ASCII file data5seis.dat. Both methods generate a set of realizations and the corresponding statistical estimators of the variables of interest (elastic properties for the seismic inversion and petrophysical properties for the geophysical inversion).

The script *FaciesClassificationDriver.py* shows the Bayesian facies classification of multiple well logs, for example, the P-wave velocity and density, assuming Gaussian distribution or nonparametric distributions estimated using kernel density estimation (Doyen, 2007). The input might include more than two variables, and the output consists of the most likely classification and the probability distributions of each facies.

## APPLICATIONS

In this section, we show six applications: (1) rock-physics model for the prediction of elastic properties and facies using well-log data, (2) geostatistical simulation of continuous and discrete properties using well logs, (3) geostatistical interpolation and simulation of 2D maps of temperature, (4) elastic inversion of partial stacked seismograms with Bayesian linearized AVO inversion, (5) rock-physics inversion of partial stacked seismograms with MCMC methods, and (6) 2D seismic inversion. All of the results were exported and plotted in MATLAB, but the Python codes exemplify the Python lines to reproduce the same plots using *matplotlib.pyplot*.

In *Application1.py*, the data set 1Ddatalog.dat includes a set of well logs from a borehole located offshore Norway. The data set includes the density, compressional, and shear sonic logs and the petrophysical curves obtained from the quantitative log interpretation of the measured logs and the corresponding depth locations of the petrophysical values. The goal of this application is to compute the rock-physics model predictions of the P- and S-wave velocity and density and the facies classification. We first load the data set and define the initial parameters of the rock-physics model. We adopt the stiff sand model (Dvorkin et al., 2014), and the initial parameters include the critical porosity, coordination number, and pressure. We assume two mineralogical components, namely, quartz and clay, and two fluid components, namely, water and oil. We compute the effective solid and fluid phase properties using the function *MatrixFluidModel*. The density of the saturated rock is computed with the function *DensityModel*. The P- and S-wave velocities are computed using the function *StiffsandModel* that combines the Hertz-Mindlin and Hashin-Shtrikman equations with the Gassmann's equations. The elastic model predictions are shown in Figure 1. We then define two facies, namely, sand and shale, and we compute a log-facies classification using a Gaussian distribution for the P- and S-wave velocity and density in each facies. The classification is obtained by applying the function *BayesGaussFaciesClass* that computes the most-likely facies and the facies probabilities. The predicted facies profile is shown in Figure 1.

In *Application2.py*, we adopt the same data set as in the first application and we select a subsample of 10 measurements of porosity and P-wave velocity. The goal of this application is to predict the porosity and P-wave velocity profiles using geostatistical interpolation and simulations. Porosity and P-wave velocity are assumed to be independent for simplicity, and interpolation and simulations are performed independently. We first interpolate the porosity measurements using the function *SimpleKriging* that computes the simple kriging estimate at each location along the well log. We then simulate 10 realizations of porosity using the function *SeqGaussianSimulation*. We then repeat the same exercise with the P-wave velocity. The kriging estimates and the stochastic realizations of the porosity and P-wave velocity are shown in Figure 2. The kriging results are generally smooth because kriging is an interpolation method that minimizes the error in the mean square sense. The SGS results show a larger heterogeneity because SGS aims to reproduce the spatial variability of the model variables. This application also includes the simulation of facies using the Markov chain model. We assume two facies, sand and shale, and we adopt a stationary first-order Markov model. We first estimate the transition matrix from the reference facies classification by counting the number of transitions. We then generate stochastic realizations using the function *MarkovChainSimulation*. Ten random realizations are shown in Figure 2.

In *Application3.py*, we apply simple kriging and SGS methods to the temperature data set measured in an area of the Yellowstone National Park. The true measurements are shown in Figure 3. From the full data set, we extract 100 temperature measurements at random locations, and we assume that this subset represents all of the available measured data. We then apply simple kriging (*SimpleKriging*) to interpolate the measurements assuming an isotropic exponential correlation function with a correlation length of 25 m and a prior Gaussian distribution with the prior mean and variance estimated from the measured data set. We then apply *SeqGaussianSimulation* to simulate

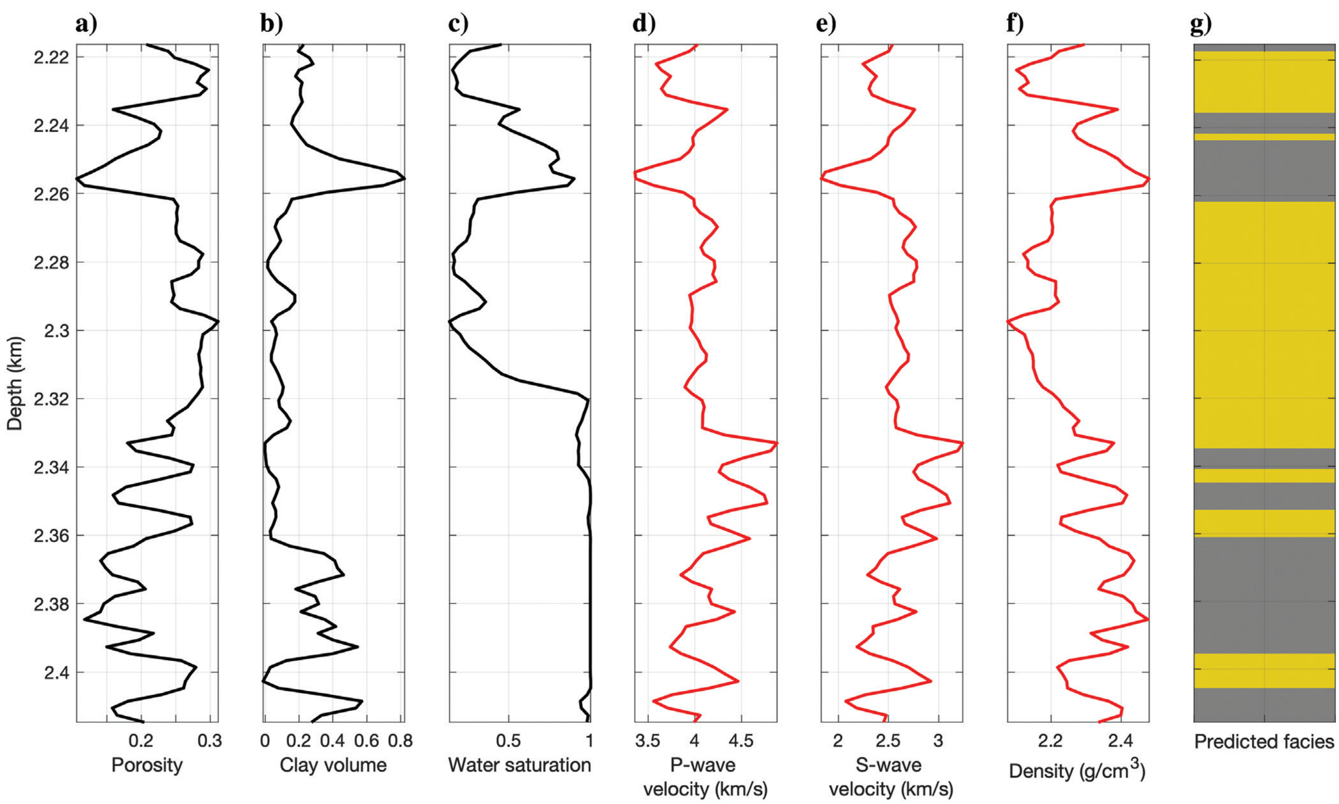


Figure 1. Application 1 — rock physics and facies classification: (a) porosity, (b) clay volume, (c) water saturation, (d) predicted P-wave velocity, (e) predicted S-wave velocity, (f) predicted density, and (g) predicted facies (sand in yellow and shale in gray).

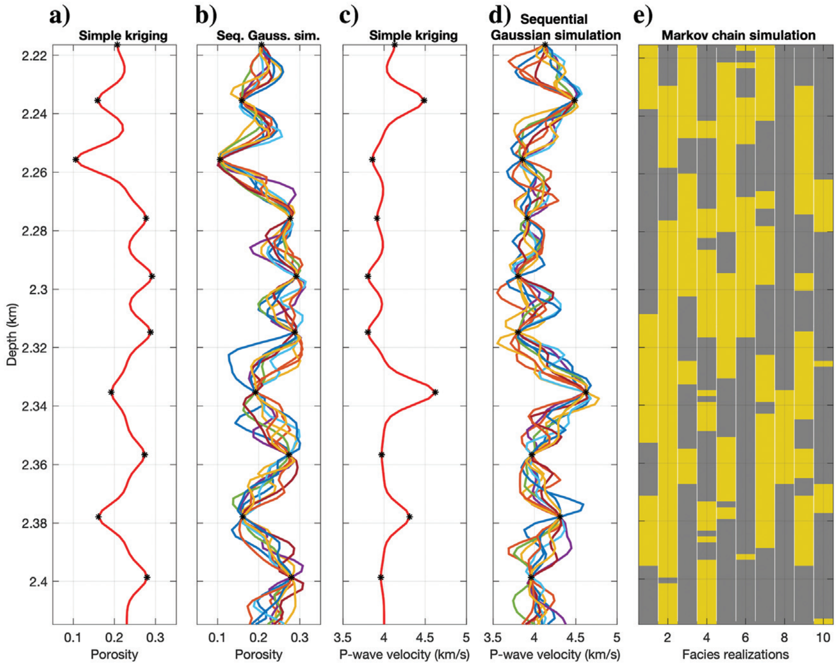


Figure 2. Application 2 — kriging and SGS: (a) interpolated porosity, (b) stochastic realizations of porosity, (c) interpolated P-wave velocity, (d) stochastic realizations of P-wave velocity, and (e) stochastic realizations of facies (sand in yellow and shale in gray). The black stars represent the actual measurements.

stochastic realizations of the temperature according to the same assumptions of simple kriging. The results are shown in Figure 3. The kriging map shows the smooth variations of the temperature, whereas the random realization shows the higher local variability.

In *Application4.py*, we apply the Bayesian linearized AVO inversion (Buland and Omre, 2003) to a synthetic seismic data set (1Ddataseis.dat) calculated from a real set of elastic well logs (1Ddatalog.dat). The data include three partial-stacked seismograms corresponding to the near, mid, and far angles (Figure 4), the two-way traveltime, and the elastic logs in the time domain. We first define the initial parameters of the model, the error distribution, and the wavelet. We generate a prior model by filtering the true elastic logs to a low frequency using the Butterworth filter. We define the spatial correlation matrix using an exponential model with correlation length of five time samples, equivalent to 5 ms. We then compute the seismic inversion results, namely, the PDF of P- and S-wave velocity and density, using the function *SeismicInversion* that returns the pointwise maximum a posteriori of the posterior distribution and the lower and upper bounds of the 0.95 confidence interval for each of the model variables. The pointwise maximum a posteriori and the 0.95 confidence interval are shown in Figure 4.

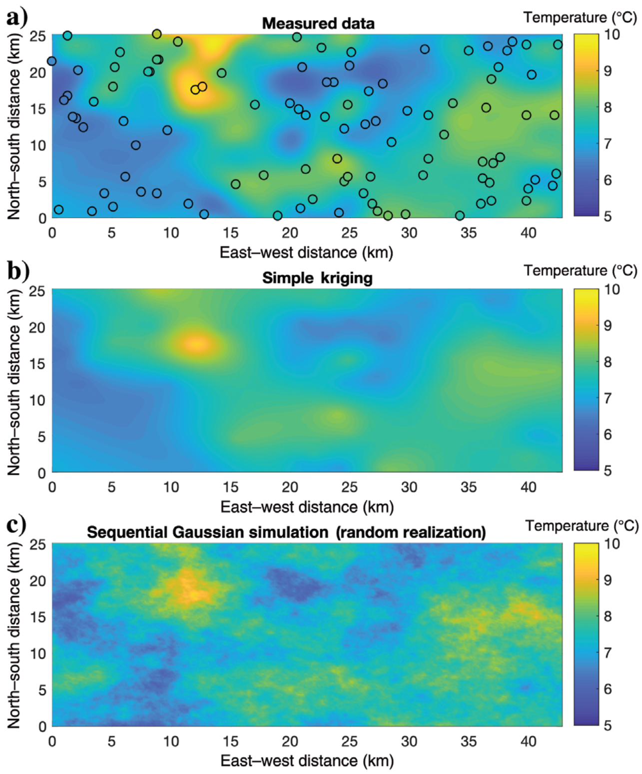


Figure 3. Application 3 — kriging and SGS of 2D data: (a) true temperature maps (the black dots represent the locations of the measured data used for interpolation and simulations), (b) simple kriging interpolation of temperature, and (c) one random stochastic realization of temperature.

In *Application5.py*, we apply the MCMC method to a synthetic seismic data set (1Ddatabseis.dat) calculated from a real set of petrophysical well logs of porosity and water saturation saved in the data set (1Ddatablog.dat) using rock physics and seismic models. The data include three partial-stacked seismograms corresponding to the near, mid, and far angles (Figure 5), the two-way traveltime, and the petrophysical logs of porosity and water saturation in the time domain. We assume that the mineral phase is homogeneous and corresponds to quartz. The error model corresponds to a signal-to-noise ratio of five. We then apply an MCMC method based on the Metropolis approach (equation 8) using the seismic forward model in *SeismicModel* for the evaluation of the likelihood. The MCMC includes $10^5$ iterations, and the posterior distribution is estimated from 90,000 realizations. Figure 5 shows a subset of 9000 realizations as well as the mean, the pointwise maximum a posteriori, and the 0.95 confidence interval of the posterior distribution. The results are overall satisfactory except for the prediction of water saturation in the bottom part of the interval, where the true data are highly skewed toward 1.

In *Application6.py*, we apply the Bayesian linearized AVO inversion to synthetic seismic data saved in the data set (2Ddataseis.dat) including three partial-stacked seismograms corresponding to the near, mid, and far angles (Figure 6). The true model used to generate the synthetic data is also provided in the data set (2Ddataelas.dat). The proposed application is a trace-by-trace inversion in which the Bayesian linearized AVO inversion approach is applied sequentially to each seismic trace. The pointwise maximum a posteriori of the posterior distribution of P- and S-wave velocity and density is shown in Figure 6.

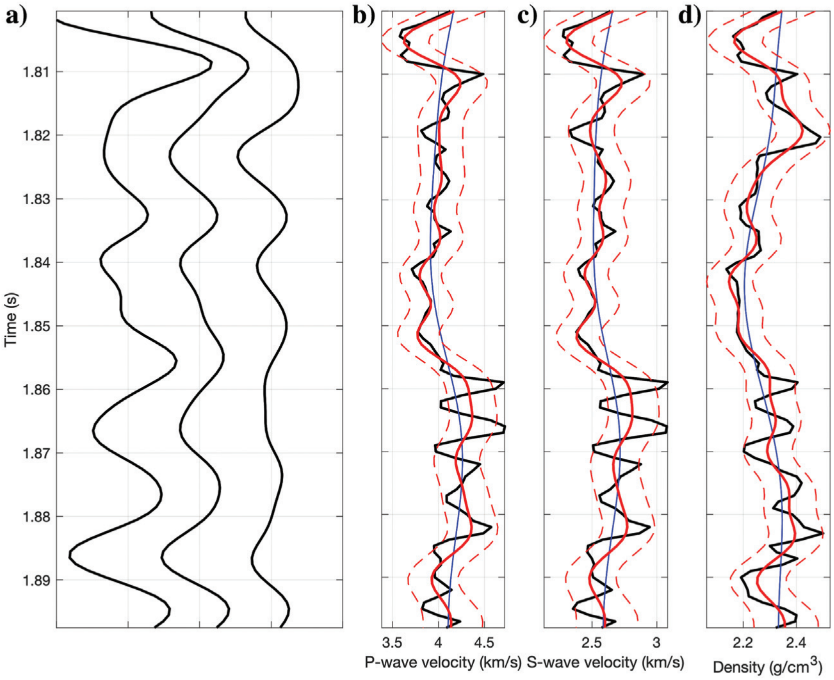


Figure 4. Application 4 — Bayesian linearized AVO seismic inversion: (a) three partial-stacked seismic data and (b-d) posterior distribution of the P- and S-wave velocity and density. The black curves represent the measured data, the blue curves represent the prior (low-frequency) model, and the red curves represent the posterior distribution (the solid red lines represent the maximum a posteriori and the dashed red lines represent the 0.95 confidence interval).

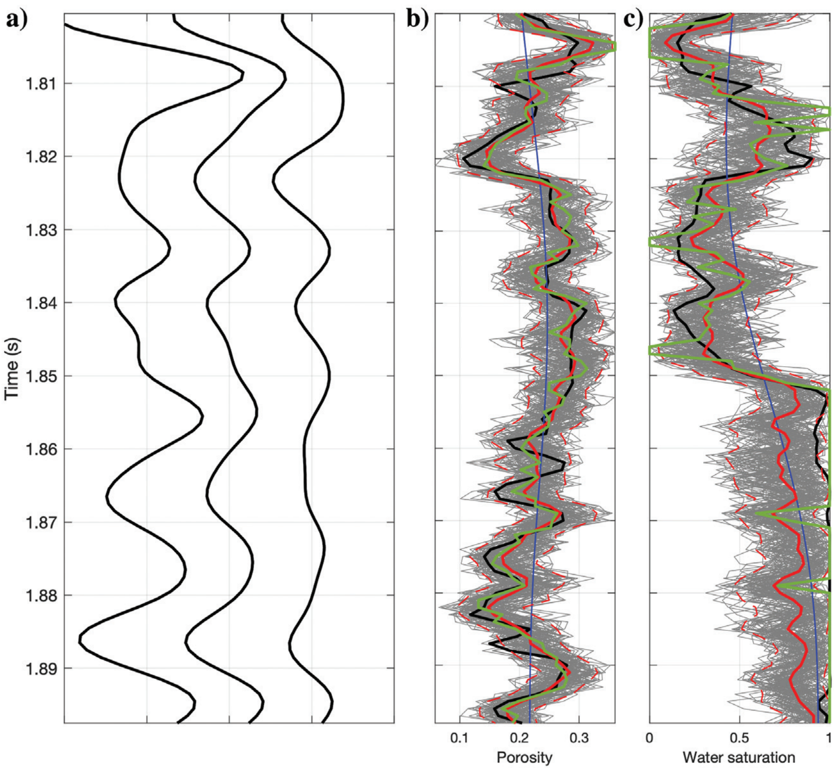


Figure 5. Application 5 – MCMC petrophysical inversion: (a) three partial-stacked seismic data and (b and c) posterior distribution porosity and water saturation. The black curves represent the measured data, the blue curves represent the prior (low-frequency) model, the red curves represent the posterior distribution (the solid red lines represent the mean, the green lines represent the maximum a posteriori model, and the dashed red lines represent the 0.95 confidence interval), and the gray curves represent randomly selected stochastic realizations.

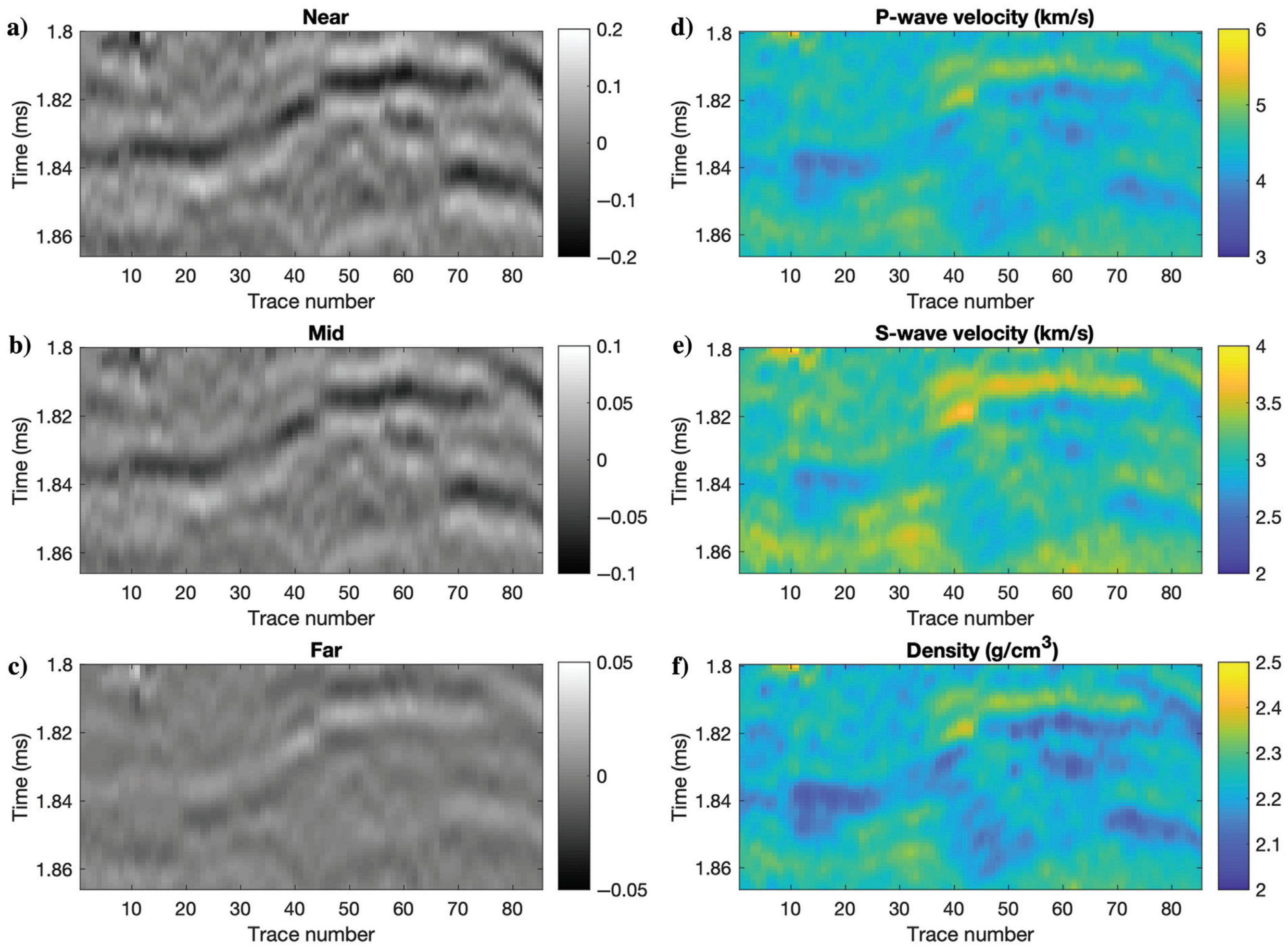


Figure 6. Application 6 — trace-by-trace 2D Bayesian linearized AVO inversion. Plots (a-c) show three partial-stacked seismic data. Plots (d-f) show the maximum a posteriori of the P- and S-wave velocity and density.

## CONCLUSION

We presented a comprehensive seismic reservoir modeling library, namely, SeReMpy. The code is written in Python and relies on four main libraries for rock physics, geostatistics, inversion, and facies classification. The SeReMpy library is the first open-source Python library that combines geophysical modeling with geostatistical methods for the prediction and stochastic simulation of reservoir properties conditioned on seismic data. The functions in the rock physics and geostatistics modules implement well known concepts and tools in geoscience such as rock-physics models for the prediction of elastic properties and geostatistical interpolation and simulation of spatially correlated random variables. These functions are then integrated in the module with recent advances in Bayesian methods for seismic and petrophysical inversion. The SeReMpy library also includes analytical solutions of the Bayesian linear Gaussian inversion problem and numerical solutions using MCMC or ensemble-based methods. The library includes several scripts that provide a tutorial for the use of the modules and multiple applications to synthetic and real data sets.

## ACKNOWLEDGMENTS

The authors acknowledge the School of Energy Resources at the University of Wyoming, the Nielson Energy Fellowship, LTrace, and the BP sponsor of the Bayesian Learning Consortium. The authors also acknowledge Alessandro Amato del Monte (Eni), Erick Talarico (Petrobras), and Mingliang Liu (Stanford University) for their contribution to the code.

## DATA AND MATERIALS AVAILBILITY

The open-source library SeReMpy is available in the GitHub repository https://github.com/seismicreservoirmodeling/SeReMpy.

## REFERENCES

Avseth, P., T. Mukerji, and G. Mavko, 2010, Quantitative seismic interpretation: Applying rock physics tools to reduce interpretation risk: Cambridge University Press.
Azevedo, L., R. Nunes, P. Correia, A. Soares, L. Guerreiro, and G. S. Neto, 2013, Multidimensional scaling for the evaluation of a geostatistical seismic

elastic inversion methodology: Geophysics, **79**, no. 5, M1–M10, doi: 10.1190/geo2013-0037.1.
Bortoli, L. J., F. A. Alabert, A. Haas, and A. G. Journel, 1993, Constraining stochastic images to seismic data: Proceedings of the 4th International Geostatistical Congress, 325–337.
Bosch, M., 1999, Lithologic tomography: From plural geophysical data to lithology estimation: Journal of Geophysical Research, **104**, 749–766, doi: 10.1029/1998JB900014.
Bosch, M., T. Mukerji, and E. F. González, 2010, Seismic inversion for reservoir properties combining statistical rock physics and geostatistics: A review: Geophysics, **75**, no. 5, 75A165–75A176, doi: 10.1190/1.3478209.
Buland, A., O. Kolbjørnsen, R. Hauge, Ø. Skjæveland, and K. Duffaut, 2008, Bayesian lithology and fluid prediction from seismic prestack data: Geophysics, **73**, no. 3, C13–C21, doi: 10.1190/1.2842150.
Buland, A., and H. Omre, 2003, Bayesian linearized AVO inversion: Geophysics, **68**, 185–198, doi: 10.1190/1.1543206.
Coléou, T., F. Allo, R. Bornard, J. Hamman, and D. Caldwell, 2005, Petrophysical seismic inversion: 75th Annual International Meeting, SEG, Expanded Abstracts, 1355–1358, doi: 10.1190/1.2147938.
Connolly, P., and M. Kemper, 2007, Statistical uncertainty of seismic net pay estimations: The Leading Edge, **26**, 1284–1289, doi: 10.1190/1.2794387.
Connolly, P. A., and M. J. Hughes, 2016, Stochastic inversion by matching to large numbers of pseudo-wells: Geophysics, **81**, no. 2, M7–M22, doi: 10.1190/geo2015-0348.1.
Contreras, A., C. Torres-Verdin, K. Kvien, T. Fasnacht, and W. Chesters, 2005, AVA stochastic inversion of pre-stack seismic data and well logs for 3D reservoir modeling: 67th Annual International Conference and Exhibition, EAGE, Extended Abstracts, cp-1-00310, doi: 10.3997/2214-4609-pdb.1.F014.
de Figueiredo, L. P., D. Grana, F. L. Bordignon, M. Santos, M. Roisenberg, and B. B. Rodrigues, 2018, Joint Bayesian inversion based on rock-physics prior modeling for the estimation of spatially correlated reservoir properties: Geophysics, **83**, no. 5, M49–M61, doi: 10.1190/geo2017-0463.1.
de Figueiredo, L. P., D. Grana, M. Roisenberg, and B. B. Rodrigues, 2019a, Gaussian mixture Markov chain Monte Carlo method for linear seismic inversion: Geophysics, **84**, no. 3, R463–R476, doi: 10.1190/geo2018-0529.1.
de Figueiredo, L. P., D. Grana, M. Roisenberg, and B. B. Rodrigues, 2019b, Multimodal Markov chain Monte Carlo method for nonlinear petrophysical seismic inversion: Geophysics, **84**, no. 5, M1–M13, doi: 10.1190/geo2018-0839.1.
Deutsch, C. V., 2002, Geostatistical reservoir modelling: Oxford University Press.
Deutsch, C. V., and A. G. Journel, 1998, GSLIB: Geostatistical software library and user's guide: Oxford University Press.
Doyen, P., 2007, Seismic reservoir characterization: EAGE.
Doyen, P. M., 1988, Porosity from seismic data: A geostatistical approach: Geophysics, **53**, 1263–1275, doi: 10.1190/1.1442404.
Dvorkin, J., M. A. Gutierrez, and D. Grana, 2014, Seismic reflections of rock properties: Cambridge University Press.
Eidsvik, J., P. Avseth, H. Omre, T. Mukerji, and G. Mavko, 2004, Stochastic reservoir characterization using prestack seismic data: Geophysics, **69**, 978–993, doi: 10.1190/1.1778241.
Emerick, A. A., and A. C. Reynolds, 2013, Ensemble smoother with multiple data assimilation: Computers & Geosciences, **55**, 3–15, doi: 10.1016/j.cageo.2012.03.011.
Fjeldstad, T., and D. Grana, 2018, Joint probabilistic petrophysics-seismic inversion based on Gaussian mixture and Markov chain prior models: Geophysics, **83**, no. 1, R31–R42, doi: 10.1190/geo2017-0239.1.
González, E. F., T. Mukerji, and G. Mavko, 2008, Seismic inversion combining rock physics and multiple-point geostatistics: Geophysics, **73**, no. 1, R11–R21, doi: 10.1190/1.2803748.
Grana, D., 2016, Bayesian linearized rock-physics inversion: Geophysics, **81**, no. 6, D625–D641, doi: 10.1190/geo2016-0161.1.
Grana, D., 2018, Joint facies and reservoir properties inversion: Geophysics, **83**, no. 3, M15–M24, doi: 10.1190/geo2017-0670.1.
Grana, D., and E. Della Rossa, 2010, Probabilistic petrophysical-properties estimation integrating statistical rock physics with seismic inversion: Geophysics, **75**, no. 3, O21–O37, doi: 10.1190/1.3386676.
Grana, D., T. Fjeldstad, and H. Omre, 2017, Bayesian Gaussian mixture linear inversion for geophysical inverse problems: Mathematical Geosciences, **49**, 1–23, doi: 10.1007/s11004-016-9671-9.
Grana, D., T. Mukerji, and P. Doyen, 2021, Seismic reservoir modelling: Wiley.
Gunning, J., and M. E. Glinsky, 2004, Delivery: An open-source model-based Bayesian seismic inversion program: Computers & Geosciences, **30**, 619–636, doi: 10.1016/j.cageo.2003.10.013.
Gunning, J., and M. E. Glinsky, 2007, Detection of reservoir quality using Bayesian seismic inversion: Geophysics, **72**, no. 3, R37–R49, doi: 10.1190/1.2713043.
Haas, A., and O. Dubrule, 1994, Geostatistical inversion — A sequential method of stochastic reservoir modeling constrained by seismic data: First Break, **12**, 561–569, doi: 10.3997/1365-2397.1994034.
Hansen, T. M., 2004, mGstat: A geostatistical matlab toolbox, Online web resource, http://mgstat.sourceforge.net, accessed 6 January 2011.
Jeong, C., T. Mukerji, and G. Mariethoz, 2017, A fast approximation for seismic inverse modeling: Adaptive spatial resampling: Mathematical Geosciences, **49**, no. 7, 845–869.
Journel, A. G., and C. J. Huijbregts, 1978, Mining geostatistics: Academic Press.
Jullum, M., and O. Kolbjørnsen, 2016, A Gaussian based framework for Bayesian inversion of geophysical data to rock properties: Geophysics, **81**, no. 3, R75–R87, doi: 10.1190/geo2015-0314.1.
Kemper, M., and J. Gunning, 2014, Joint impedance and facies inversion — Seismic inversion redefined: First Break, **32**, 89–95, doi: 10.3997/1365-2397.32.9.77968.
Larsen, A. L., M. Ulvmoen, H. Omre, and A. Buland, 2006, Bayesian lithology/fluid prediction and simulation on the basis of a Markov-chain prior model: Geophysics, **71**, no. 5, R69–R78, doi: 10.1190/1.2245469.
Liu, M., and D. Grana, 2018, Stochastic nonlinear inversion of seismic data for the estimation of petroelastic properties using the ensemble smoother and data reparameterization: Geophysics, **83**, no. 3, M25–M39, doi: 10.1190/geo2017-0713.1.
Liu, M., and D. Grana, 2019, Accelerating geostatistical seismic inversion using TensorFlow: A heterogeneous distributed deep learning framework: Computers & Geosciences, **124**, 37–45, doi: 10.1016/j.cageo.2018.12.007.
Mavko, G., T. Mukerji, and J. Dvorkin, 2020, The rock physics handbook: Cambridge University Press.
Mosegaard, K., and A. Tarantola, 1995, Monte Carlo sampling of solutions to inverse problems: Journal of Geophysical Research, **100**, 12431–12447, doi: 10.1029/94JB03097.
Mukerji, T., A. Jørstad, P. Avseth, G. Mavko, and J. R. Granli, 2001, Mapping lithofacies and pore-fluid probabilities in a North Sea reservoir: Seismic inversions and statistical rock physics: Geophysics, **66**, 988–1001, doi: 10.1190/1.1487078.
Pebesma, E. J., 2004, Multivariable geostatistics in S: The gstat package: Computers & Geosciences, **30**, 683–691, doi: 10.1016/j.cageo.2004.03.012.
Remy, N., A. Boucher, and J. Wu, 2009, Applied geostatistics with SGeMS: A user's guide: Cambridge University Press.
Rimstad, K., P. Avseth, and H. Omre, 2012, Hierarchical Bayesian lithology/fluid prediction: A North Sea case study: Geophysics, **77**, no. 2, B69–B85, doi: 10.1190/geo2011-0202.1.
Sambridge, M., and K. Mosegaard, 2002, Monte Carlo methods in geophysical inverse problems: Reviews of Geophysics, **40**, 3.1–3.29, doi: 10.1029/2000RG000089.
Scales, J. A., and L. Tenorio, 2001, Prior information and uncertainty in inverse problems: Geophysics, **66**, 389–397, doi: 10.1190/1.1444930.
Sen, M., and P. Stoffa, 1996, Bayesian inference, Gibbs sampler and uncertainty estimation in geophysical inversion: Geophysical Prospecting, **44**, 313–350, doi: 10.1111/j.1365-2478.1996.tb00152.x.
Spikes, K., T. Mukerji, J. Dvorkin, and G. Mavko, 2008, Probabilistic seismic inversion based on rock-physics models: Geophysics, **72**, no. 5, R87–R97, doi: 10.1190/1.2760162.
Tarantola, A., 2005, Inverse problem theory: SIAM.
Tarantola, A., and B. Valette, 1982, Inverse problems = quest for information: Journal of Geophysics, **50**, 159–170.
Ulrych, T. J., M. D. Sacchi, and A. Woodbury, 2001, A Bayes tour of inversion: A tutorial: Geophysics, **66**, 55–69, doi: 10.1190/1.1444923.
Ulvmoen, M., and H. Omre, 2010, Improved resolution in Bayesian lithology/fluid inversion from prestack seismic data and well observations — Part 1: Methodology: Geophysics, **75**, no. 2, R21–R35, doi: 10.1190/1.3294570.

Biographies and photographs of the authors are not available.