GEOPHYSICS, VOL. 85, NO. 4 (JULY-AUGUST 2020); P. WA279–WA292, 16 FIGS., 5 TABLES.
10.1190/GEO2019-0200.1

The relevance vector machine for seismic Bayesian compressive sensing

Georgios Pilikos[1]

ABSTRACT

Missing traces in seismic surveys create gaps in the data and cause problems in later stages of the seismic processing workflow through aliasing or incoherent noise. Compressive sensing (CS) is a framework that encompasses data reconstruction algorithms and acquisition processes. However, CS algorithms are mainly ad hoc by focusing on data reconstruction without any uncertainty quantification or feature learning. To avoid ad hoc algorithms, a probabilistic data-driven model is used, the relevance vector machine (RVM), to reconstruct seismic data and simultaneously quantify uncertainty. Modeling of sparsity is achieved using dictionaries of basis functions, and the model remains flexible by adding or removing them iteratively. Random irregular sampling with time-slice processing is used to reconstruct data without aliasing. Experiments on synthetic and field data sets illustrate its effectiveness with state-of-the-art reconstruction accuracy. In addition, a hybrid approach is used in which the domain of operation is smaller while, simultaneously, learned dictionaries of basis functions from seismic data are used. Furthermore, the uncertainty in predictions is quantified using the predictive variance of the RVM, obtaining high uncertainty when the reconstruction accuracy is low and vice versa. This could be used for the evaluation of source/receiver configurations guiding seismic survey design.

INTRODUCTION

A seismic survey acquires data and allows processing and imaging that leads to an image of the interior structure of the earth. An artificial source of body waves is used at the surface. This creates reflections from deep impedance changes at rock layer boundaries, and receivers record these reflections. Nevertheless, receivers or sources, occasionally, cannot be placed in restricted locations. Being able to obtain seismic data from fewer receivers and sources without significantly compromising their quality is not only essential when they are missing, but it has great importance for environmental, economic, and safety reasons. It is also an important part of seismic data processing because it benefits other steps such as migration.

Seismic compressive sensing (CS) is a framework that encompasses data reconstruction algorithms and also guides the data acquisition process. The data reconstruction aspect uses the assumption of sparsity for seismic data. Seismic events are assumed to be sparse in some transforms such as the Fourier (Sacchi et al., 1998; Gülünay, 2003; Liu and Sacchi, 2004; Xu et al., 2005; Abma and Kabir, 2006), the Radon (Kabir and Verschuur, 1995; Trad et al., 2002), the curvelet (Hennenfent and Herrmann, 2008; Herrmann and Hennenfent, 2008; Naghizadeh and Sacchi, 2010a; Shahidi et al., 2013), the focal (Kutscha and Verschuur, 2016), the seislet (Fomel and Liu, 2010; Liu and Fomel, 2010; Gan et al., 2015, 2016), and the shearlet (Kong and Peng, 2015) transforms. These dictionaries of basis functions are used in conjunction with a sparse solver to obtain a solution.

In particular, projection onto convex sets (POCS) (Abma and Kabir, 2006) can transform the data to any domain, more traditionally to the Fourier domain, and it uses hard or soft thresholding (Stanton et al., 2015) when choosing which components to keep in the solution. Another solver such as iteratively reweighted least-squares was also proposed (Zwartjes and Sacchi, 2007) to suppress the artifacts in the Fourier domain. For the curvelet transform, the iterative soft thresholding (IST) approach (Herrmann and Hennenfent, 2008) has been suggested. A faster version of IST was proposed (Beck and Teboulle, 2009), namely, the fast iterative soft thresholding algorithm, and then it was applied to seismic data (Pérez et al., 2013). Other techniques include reconstruction using wavefront attributes (Gajewski and Xie, 2017), hybrid sparsity-rank constraint (Zhang et al., 2017), and damped rank-reduction (Chen et al., 2016). In a comparison of interpolators, POCS was found to better preserve the amplitudes of seismic data (Stanton et al., 2012).

A different approach is to solve the l_1-norm minimization problem. The spectral projected gradient for L1 (SPGL1) (van den Berg

Manuscript received by the Editor 2 April 2019; revised manuscript received 2 April 2020; published ahead of production 2 June 2020; published online 24 June 2020.
[1]Formerly University of Cambridge, Department of Physics, Laboratory for Scientific Computing, Maxwell Centre, Cambridge, United Kingdom. Presently Computational Imaging Group at Centrum Wiskunde & Informatica, Amsterdam, Netherlands. E-mail: Georgios.Pilikos@cwi.nl (corresponding author).

and Friedlander, 2009) was proposed to solve this and is used in the literature obtaining state-of-the-art results with various dictionaries (Jingjie et al., 2015; Kutscha and Verschuur, 2016). Other methods of seismic data reconstruction include matrix and tensor reconstruction (Trickett et al., 2010; Oropeza and Sacchi, 2011; Kreimer and Sacchi, 2012; Gao et al., 2013; Kumar et al., 2015) via rank and nuclear norm minimization. These rank reduction solvers assume that missing traces and noise increase the rank of the data matrix and thus try to minimize it.

An alternative to the predefined dictionaries would be to learn the basis functions from the available seismic data. This approach was used by Zhu et al. (2015) for the purpose of denoising seismic data. The main algorithm is a modification of the K-singular value decomposition (Elad and Aharon, 2006), which alternates between optimizing the coefficients and the dictionary, for the given data. Furthermore, simultaneous denoising and feature learning of seismic data were performed by Beckouche and Ma (2014), and further dictionary learning for denoising was undertaken by Turquais et al. (2015). Another approach is to use a data-driven tight frame and learn a set of filters (features/basis) to sparsely represent the seismic data (Liang et al., 2014) obtaining results similar to POCS. Yu et al. (2015) extend this method for high-dimensional seismic data with great reconstruction accuracy but high computational cost. Yu et al. (2016) use fewer patches during training by carefully selecting optimum patches depending on their variance to speed up the process. An alternative to speeding up the learning process using tight frames was studied by Siahsar et al. (2017) who use a nonnegativity constraint to reduce the space of the solution, consequently decreasing the computational cost and boosting sparsity in data representation.

Recently, machine-learning techniques have been introduced to predict missing receivers' values. Support vector regression (SVR) (Jia and Ma, 2017) has been used for seismic data reconstruction by learning a hyperplane that describes the relationship between input and output data. A faster version has been proposed by Jia et al. (2018), which uses a subset of the training data using Monte Carlo machine learning. A probabilistic version of SVR is the relevance vector machine (RVM) (Tipping, 2001), which again learns a function of inputs to outputs but is built around a probabilistic framework. This was used by Pilikos and Faul (2016) for seismic data reconstruction and obtained better results compared with POCS and SPGL1. Another machine-learning technique is beta process factor analysis (BPFA) (Paisley and Carin, 2009; Zhou et al., 2012), which learns dictionaries of basis from seismic data with success (Pilikos and Faul, 2017) and has no signs of aliasing in the frequency-wavenumber (f-k) domain (Pilikos et al., 2017). The latter two techniques are based around a Bayesian CS (Ji et al., 2008) framework. An excellent introduction to the Bayesian inference field for seismic data is given by Duijndam (1988a, 1988b) and Ulrych et al. (2001).

Contributions

Within this framework, we propose to use the RVM to reconstruct undersampled data and simultaneously quantify uncertainty in predictions. Reconstruction accuracy and computational speed analysis are performed on synthetic data. The main domain used in the experiments is the time slice over all time steps in the data set. Note that the reconstruction is done in this domain, but then the data are resorted in shot gathers as in traditional seismic data reconstruction. We also use irregular undersampling to avoid aliasing in the f-k domain (Kumar et al., 2015). In particular, random placement of receivers is used, and we show why this is advantageous for the RVM. This arrangement of receivers/sources is not currently used in the field, but if high reconstruction accuracy can be achieved, the acquisition design process could change in the future. Furthermore, we use the Bayesian inference framework and obtain uncertainties for predictions. We vary the configuration of receivers, and we obtain uncertainty maps and illustrate that the uncertainty of the model is low when the reconstruction accuracy is high and vice versa.

In addition, we propose to use a hybrid approach. The domain of operation is reduced, and at the same time learned dictionaries of basis functions for seismic data are used by the RVM and SPGL1. These approaches are referred to as RVM-learned and SPGL1-learned, respectively, and we illustrate the improved reconstruction accuracy achieved while using small patch sizes. The f-k domain analysis demonstrates that the reconstructions can be achieved without any signs of aliasing. We also provide analysis on different parts of the signal and propose different configurations of algorithms for improved overall reconstruction accuracy.

Notation

Bold lowercase letters such as $\mathbf{w}$ represent column vectors, bold uppercase letters such as $\mathbf{\Phi}$ represent matrices, and $\|.\|$ represents the norm of a vector or a matrix. The superscript (i) (e.g., in $t^{(i)}$) corresponds to a certain sample in a trace.

THE RVM

The RVM is a probabilistic data-driven model that assumes that data are generated by a linear combination of predefined nonlinear basis functions. For a collection of available traces N, the model is defined by

$$t^{(i)} = \sum_{l=1}^{L} w_l \boldsymbol{\phi}_l(\mathbf{k}^{(i)}) + \epsilon^{(i)} = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{k}^{(i)}) + \epsilon^{(i)}, \quad (1)$$

where $t^{(i)} \in \mathbb{R}$ is a trace sample, $\mathbf{k}^{(i)} \in \mathbb{R}^c$ are the coordinates, $\mathbf{w} \in \mathbb{R}^L$ are the coefficients of the linear combination of the transformed input data, $\epsilon^{(i)} \sim \mathcal{N}(0, \sigma^2)$ is independent and identically distributed (iid) additive noise, and $\boldsymbol{\phi}(\mathbf{k}^{(i)}) = [\boldsymbol{\phi}_1(\mathbf{k}^{(i)}), \boldsymbol{\phi}_2(\mathbf{k}^{(i)}), \ldots, \boldsymbol{\phi}_L(\mathbf{k}^{(i)})]^T \in \mathbb{R}^L$ with each entry being a certain basis function applied to a particular data point i. The corresponding likelihood function (Tipping, 2001) is given by

$$p(t^{(i)}|\mathbf{w}, \boldsymbol{\phi}(\mathbf{k}^{(i)}), \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \times \exp\left\{-\frac{1}{2\sigma^2}(t^{(i)} - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{k}^{(i)}))^2\right\}. \quad (2)$$

Because there are N samples and using the iid assumption,

$$p(\mathbf{t}|\mathbf{w}, \mathbf{\Phi}, \sigma^2) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(t^{(i)} - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{k}^{(i)}))^2\right\} \quad (3)$$

$$= \frac{1}{\left(\sqrt{2\pi\sigma^2}\right)^N} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{t}-\mathbf{\Phi}\mathbf{w}\|^2\right\}, \quad (4)$$

where $\mathbf{\Phi} \in \mathbb{R}^{N\times L}$. It is given by $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_L]$ where each $\boldsymbol{\phi}_l \in \mathbb{R}^N$ is the lth basis function evaluated at N available samples.

For CS applications, there is a required assumption that the acquired signal is sparse (or it can be transformed to a sparse domain). Thus, a prior probability distribution on the model parameters $\mathbf{w}$ that promotes sparsity is required. A normal prior distribution is preferred, which is conjugate to the likelihood function. Each coefficient w_l is associated with a different variance that is controlled by the precision α_l. Thus, the prior distribution is given by

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{l=1}^{L} \mathcal{N}(w_l; 0, \alpha_l^{-1}) = \prod_{l=1}^{L} \sqrt{\frac{\alpha_l}{2\pi}} e^{-\frac{\alpha_l}{2}w_l^2}, \quad (5)$$

which is a product of zero-mean normal distributions with each distribution having a precision α_l. These parameters scale the normal distributions according to their value. Suitable priors are the gamma distributions given by

$$p(\boldsymbol{\alpha}) = \prod_{l=1}^{L} \text{Gamma}(\alpha_l; c, d) = \prod_{l=1}^{L} \frac{1}{\Gamma(c)} d^c \alpha_l^{c-1} e^{-d\alpha_l}, \quad (6)$$

where $\Gamma(c) = \int_0^\infty t^{c-1}e^{-t}dt$ is the Gamma function, and c, d are the scale and shape parameters for the distribution. The noise precision σ^{-2} is also modeled by a gamma distribution,

$$p(\sigma^{-2}) = \text{Gamma}(\sigma^{-2}; e, f), \quad (7)$$

with e, f being the scale and shape parameters for the gamma distribution. Typically, the Gamma distributions are chosen to be flat and the parameters that control them are set to small values such as $c = d = e = f = 10^{-6}$ (Tipping, 2001). This results in a hierarchical prior formulation that provides flexibility in the coefficients. It allows some probability mass to potentially concentrate on a few coefficients and others to be zero or close to zero resulting in the desired property of sparsity. A summary of the RVM is given in the graphical model of Figure 1.

From Bayes' rule and conjugacy, we can get the posterior distribution of the model coefficients being normal with covariance Σ and mean $\boldsymbol{\mu}$ given by

$$\mathbf{\Sigma} = (\sigma^{-2}\mathbf{\Phi}^T\mathbf{\Phi} + \boldsymbol{A})^{-1}, \quad (8)$$

$$\boldsymbol{\mu} = \sigma^{-2}\mathbf{\Sigma}\mathbf{\Phi}^T\mathbf{t}, \quad (9)$$

where $\boldsymbol{A}$ is the diagonal matrix $\text{diag}(\alpha_1, \alpha_2, \ldots, \alpha_L)$. Inference involves finding the optimum configuration of $\boldsymbol{\alpha}$ and σ^2. The original algorithm (Tipping, 2001) uses a computationally intensive procedure that obtains these parameters through the inversion of the covariance matrix.

In this paper, a fast version of the RVM (Tipping and Faul, 2003) is used, summarized in Algorithm 1. There are different possibilities for this algorithm. First, the initialization of the noise variance σ^2 should be such that the noise is not overestimated. An empirical study is provided later. The choice of the first basis function is done by finding the largest normalized projection onto the available data. That is, the algorithm searches for the largest $\|\boldsymbol{\phi}_l^T\mathbf{t}\|^2/\|\boldsymbol{\phi}_l\|^2$ for all l and the basis function $\boldsymbol{\phi}_l$ with the largest projection is chosen to initialize the model. Then, this is used to calculate the first α_l as

$$\alpha_l = \frac{\|\boldsymbol{\phi}_l\|^2}{\|\boldsymbol{\phi}_l^T\mathbf{t}\|^2/\|\boldsymbol{\phi}_l\|^2 - \sigma^2}. \quad (10)$$

In subsequent iterations, the algorithm needs to choose a basis function from the dictionary and decide what to do with it. This can be done at random or from a predefined list. To find the greatest increase in the log-likelihood per iteration, a formula is given in the appendix of Tipping and Faul (2003) along with update formulas for the various options that the algorithm can take for either step 6, 7, or 8. The last consideration is the convergence criterion. If there is no significant difference in the log $\boldsymbol{\alpha}$ for all basis functions, then the algorithm terminates. A suitable threshold is proposed as 10^{-6} by Tipping and Faul (2003).

For completeness, we define two further variables that are used in Algorithm 1. The term s_j is called the sparsity factor, and it gives a

Algorithm 1. Fast RVM.

1: **procedure** FastRVM
2: Initialize σ^2, $\boldsymbol{\phi}_l$, and α_l [Details are in the text.]
3: Compute Σ and $\boldsymbol{\mu}$ and s_l, q_l $\forall\ l$
4: Choose a basis function $\boldsymbol{\phi}_l$ from the dictionary [Details are in the text.]
5: Calculate $\theta_l = q_l^2 - s_l$ [Details are in the text.]
6: **if** $(\theta_l > 0)$ and $(\alpha_l < \infty)$ **then** reestimate α_l
7: **if** $(\theta_l > 0)$ and $(\alpha_l = \infty)$ **then** add α_l
8: **if** $(\theta_l \leq 0)$ and $(\alpha_l < \infty)$ **then** delete α_l
9: If not converged, go back to 4.
10: **end.**

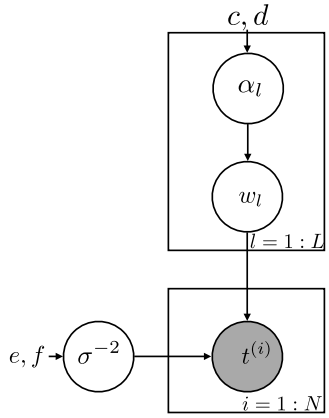

Figure 1. Graphical model of the RVM illustrating the hierarchical prior on the model parameters. The variable $t^{(i)}$ is a sample from a trace generated by a linear combination of basis functions with coefficients w_l, where each l corresponds to a basis function. The variable α_l is the hyperprior on each coefficient and controls the model sparsity. σ^{-2} is the noise precision in the model.

measure of how much the basis function $\boldsymbol{\phi}_j$ overlaps with the basis functions already in the model. It is defined as

$$s_j = (\boldsymbol{\phi}_j)^T(\boldsymbol{C}_{-j})^{-1}\boldsymbol{\phi}_j. \tag{11}$$

The term q_j is called the quality factor and is given by

$$q_j = (\boldsymbol{\phi}_j)^T(\boldsymbol{C}_{-j})^{-1}\mathbf{t}. \tag{12}$$

Both depend on $\mathbf{C}$, which is defined as

$$\mathbf{C} = \sigma^2\mathbf{I} + \sum_{l\neq j}(\alpha_l)^{-1}\boldsymbol{\phi}_l(\boldsymbol{\phi}_l)^T + (\alpha_j)^{-1}\boldsymbol{\phi}_j(\boldsymbol{\phi}_j)^T, \tag{13}$$

where $\boldsymbol{\phi}_j$ is the jth basis function applied to all available data points. Using the above, another definition is given by

$$\mathbf{C}_{-j} = \sigma^2\mathbf{I} + \sum_{l\neq j}(\alpha_l)^{-1}\boldsymbol{\phi}_l(\boldsymbol{\phi}_l)^T. \tag{14}$$

Further information about these and the RVM can be found in (Tipping and Faul, 2003).

After the algorithm terminates, it returns the statistics of the model that are $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$. These are used to create the predictive distribution $\mathcal{N}(t^{(*)}, \sigma_*^2)$ where the predictive mean is

$$t^{(*)} = m_* = \boldsymbol{\mu}^T\boldsymbol{\phi}(\mathbf{k}^{(*)}) \tag{15}$$

and the predictive variance is

$$\sigma_*^2 = v_* = \sigma^2 + \boldsymbol{\phi}(\mathbf{k}^{(*)})^T\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{k}^{(*)}), \tag{16}$$

where $\boldsymbol{\phi}(\mathbf{k}^{(*)})$ is a vector of all basis functions in the model calculated at a missing receiver $\mathbf{k}^{(*)}$. Pilikos and Faul (2016) use this for seismic CS and illustrate that the RVM with the discrete cosine transform (DCT) obtains better results than other configurations. We will examine this further, with analysis on reconstruction accuracy and uncertainty quantification.

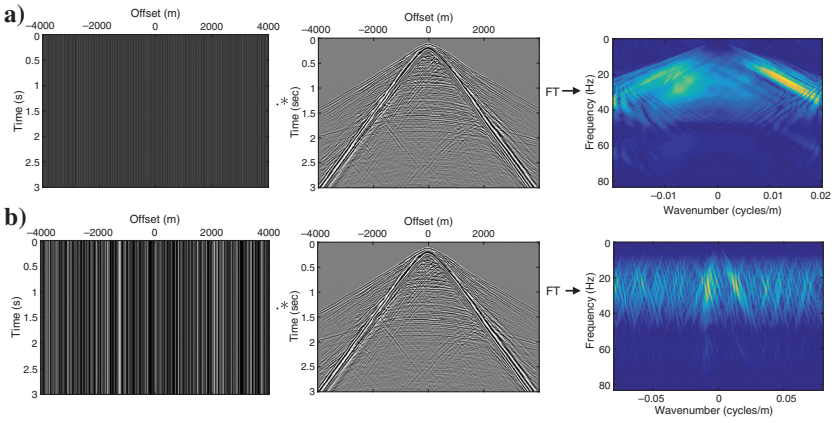

Figure 2. Regular undersampling (a) results in aliasing in the f-k domain. However, irregular undersampling (b) introduces incoherent noise in the f-k domain, but not aliasing.

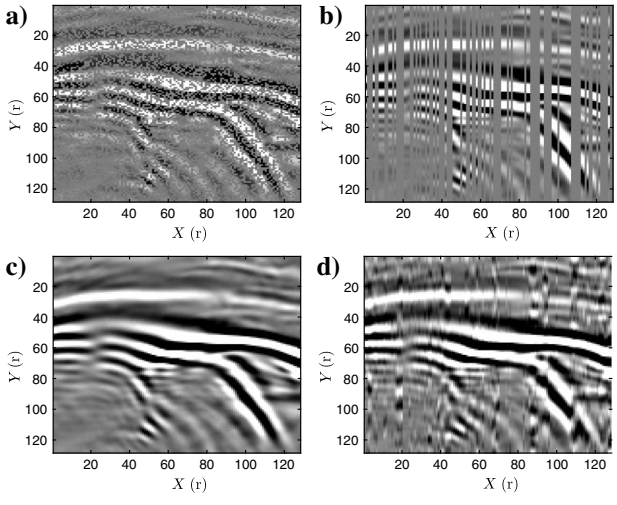

Figure 3. (a) Using 50% of the randomly selected receivers, (b) using 50% of the randomly selected receiver lines, (c) reconstruction of (a) with $Q = 47.322$ db, and (d) reconstruction of (b) with $Q = 12.524$ db. The term r represents the receiver index.

EXPERIMENTAL SETUP

To evaluate the performance of the RVM, we used a synthetic data set called SEAM-II (Oristaglio, 2012; SEG, 2018a). The data set has a 6.25 m spatial sampling. There are 1281 lines of receivers with 1281 receivers per line spanning 8000 m in each direction. The temporal sampling is 6 ms and each receiver is composed of 500 time samples resulting in 3 s of recordings. Using this data set, we will extract various time slices and shot records, all using shot gathers to test different algorithms. Note that the algorithms will be tested for common shot gathers when receivers are missing, but the same principles hold for common receiver gathers when shots are missing. To evaluate the reconstruction accuracy of all algorithms, we will use the reconstruction quality Q defined by

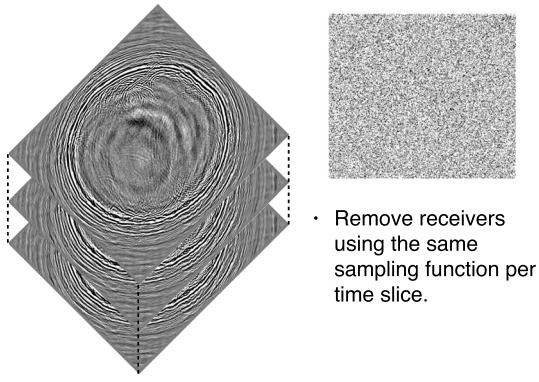

Figure 4. Time-slice processing illustration. The same sampling function is used for each time slice. Then, each time slice is used by an algorithm for processing.

$$Q[\text{db}] = 10 \log \frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}, \tag{17}$$

where $\mathbf{x}$ is the original signal and $\hat{\mathbf{x}}$ is the reconstruction. This metric was used by Kazemi and Sacchi (2014) and Kazemi et al. (2016) for seismic data, among others. It captures the error in the reconstruction by directly comparing the traces' values of the predicted and original signal. We will also use the f-k domain of the reconstructed data to visualize if there is aliasing or incoherent noise in the frequency spectrum.

Reasoning behind irregular sampling

Depending on the type of spatial undersampling, the f-k domain of a shot gather varies. If there is regular undersampling, then the seismic data could be aliased. This happens when the sampling rate (in this example, 25 m spacing, which is four times under-sampled) is not sufficient as can be seen in Figure 2a. However, with the same amount of undersampling but in random locations (irregular), the f-k domain does not show any signs of aliasing but rather incoherent noise. This can be seen in Figure 2b, and it was also discussed by various other authors such as Naghizadeh and Sacchi (2010b), Naghizadeh (2012), and Kumar et al. (2015). To avoid aliasing, but at the same time create more efficient seismic surveys, we choose to undersample seismic data using irregular (random) sampling and then reconstruct missing values using the RVM. Random sampling is not realistic for current seismic surveys, but if we do not have aliasing in reconstructions then the budget and the acquisition design of seismic surveys could change.

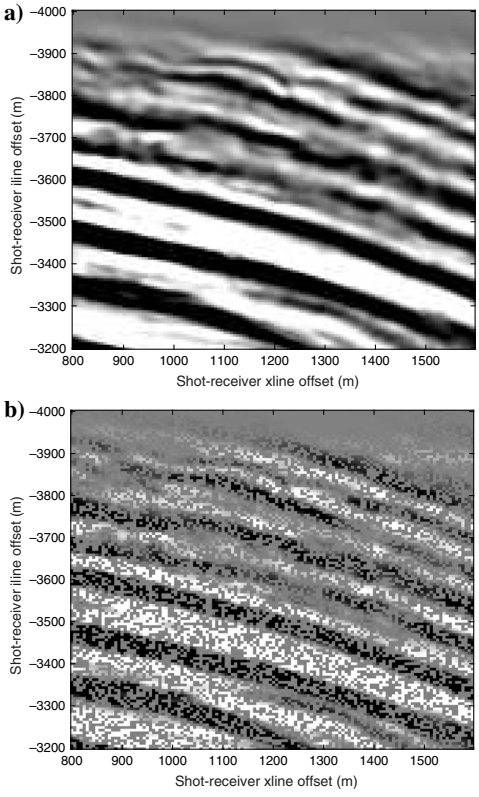

Figure 5. (a) Original section of 128 × 128 receivers extracted from a time slice and (b) the same signal using only 50% of receivers.

Reasoning behind time-slice processing

We focus on analysing the data in two domains: the x-t domain (shot record) and the x-y domain (time slice). Missing receivers in each domain correspond to different patterns of removal. We draw a number corresponding to the percentage of receivers, from the uniform distribution between 1 and 100. If it is less than or equal to the percentage used, the sampling function corresponding to the receiver is set to 1, or otherwise to 0. We use the RVM to reconstruct a section of a time slice with the same percentage of receivers used but using two different sampling functions. Figure 3a shows the seismic time slice used for the experiment by the RVM with the sampling function for time slices for 50% of receivers randomly selected. Figure 3b shows the same signal but using the sampling function for the x-t domain for 50% of receiver lines randomly selected. Reconstructions are shown in Figure 3c and 3d. We can see that the reconstruction using the sampling function with randomly missing receivers has better reconstruction quality, $Q = 47.322$ db, as expected, as opposed to the reconstruction using the sampling function with randomly missing receiver lines with $Q = 12.524$ db.

These results are due to the fact that missing receivers in the x-t domain correspond to entire missing columns from the training

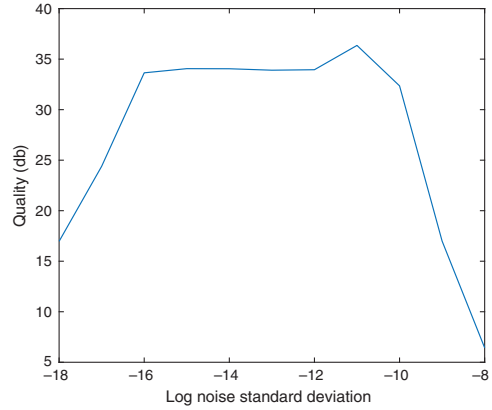

Figure 6. Plot of the reconstruction quality Q against the log noise standard deviation. When the noise standard deviation is very small or very large, the quality worsens.

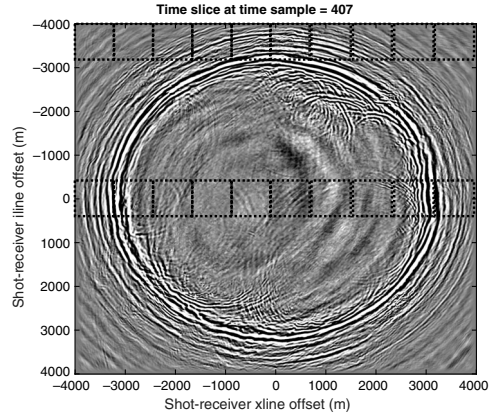

Figure 7. An example of a time slice with each square illustrating a section from a time slice close to and far from the offset.

data. For the RVM, this is a problem because there are not enough training data to learn a model that describes the variations in those regions. Therefore, in all subsequent experiments, we will be using time slices as the domain for reconstructing seismic data over all time samples and then re-sort the data back in the *x-t* domain as done in traditional seismic data reconstruction. Figure 4 illustrates this idea. Note that, for each time sample, the receivers removed are in the same locations. This results in lines missing in the *x-t* domain as done traditionally.

Empirical initialization of noise variance

From Algorithm 1, we need to initialize model parameters such as the noise variance σ^2. An empirical study is performed but more specifically for the noise standard deviation σ. In total, 11 different values for this are tested from $\sigma = 10^{-18}$ to $\sigma = 10^{-8}$, with an increment of a factor of 10. This range has been chosen because the seismic data that we use for evaluation have a standard deviation in the order of 10^{-10}. The original signal can be seen in Figure 5a, and the same signal using only 50% of the receivers can be seen in Figure 5b. We varied the noise standard deviation as mentioned, and we obtained various values for the reconstruction accuracy measured in Q as defined in equation 17. Figure 6 shows the reconstruction quality Q with different initializations of the noise standard deviation.

The log noise standard deviation is plotted for visualization convenience. It can be seen that when the noise standard deviation is very small or very large, the RVM behaves badly. If it is too small, it affects the calculation of the mean and the covariance of the model parameters because it scales them. If it is too large, the algorithm stops earlier than it should because it assumes that the remaining error is explained by the noise. There is a region where it obtains comparable reconstruction accuracy when the noise standard deviation is from 10^{-16} to 10^{-12}. Nevertheless, the reconstruction accuracy peaks at 10^{-11}, and this will be used throughout the experiments.

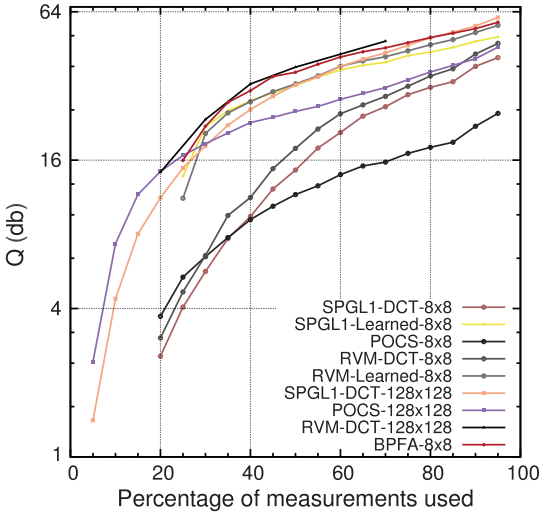

Figure 8. Mean reconstruction accuracy Q against the percentage of measurements for 150 seismic sections of time slices.

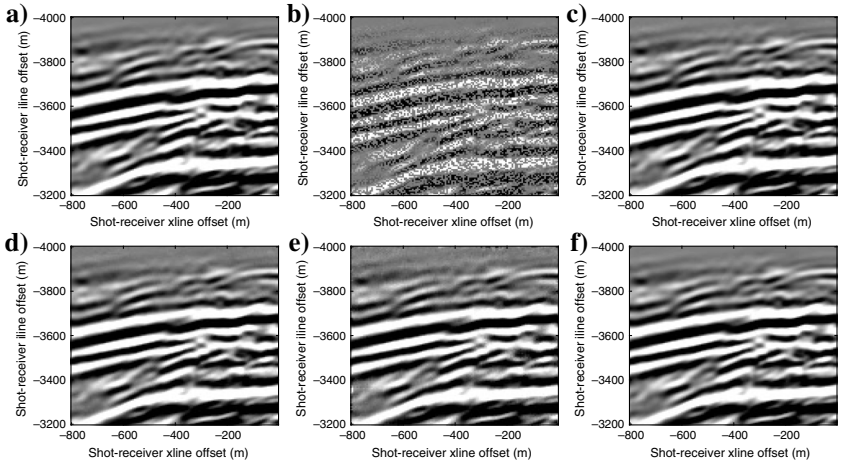

Figure 9. A section from a far-offset time slice. Reconstructions with different algorithms are included using 50% of the receivers. (a) Original, (b) 50% used, (c) RVM-DCT, 128×128, $Q =$ **42.07** db, (d) SPGL1-DCT, 128×128, $Q = 34.71$ db, (e) POCS, 128×128, $Q = 24.32$ db, and (f) BPFA, 8×8, $Q = 41.20$ db.

Parameter settings

Different configurations of the RVM produce different results (Pilikos and Faul, 2016). For comparison, we use the best configuration with regard to the reconstruction accuracy. This configuration is the RVM using DCT, which benefits from larger patch sizes (Pilikos and Faul, 2016). To compare with other algorithms that operate on smaller patch sizes such as BPFA (Pilikos and Faul, 2017), we will provide results for configurations of 128×128 and 8×8 sizes. In all experiments, SPGL1 (van den Berg and Friedlander, 2007) software with DCT is used. Similarly, for POCS, the MATLAB code (Abma, 2018) was used, and for the RVM, the package from the author's website (Tipping, 2018) is ran. Finally, for BPFA, the MATLAB package (Zhou, 2018) is used.

For POCS and SPGL1, various configurations also provide different levels of reconstruction accuracy. These could be various parameter initializations, patch sizes, basis functions, and stopping criteria to name a few. To address this variability in full, experiments with all possible configurations are necessary. However, this is not the purpose of the paper. Pilikos and Faul (2017) explore two key parameters: the patch size and the stopping criteria. Experiments showed that POCS with 500 iterations and a patch size of 128×128 obtained the best compromise between time and accuracy. For the stopping criterion of SPGL1, the value of the residual should have a difference that is much smaller than the l_2 norm of the available data, e.g., between $10^{-6}\|\mathbf{x}\|_2$ and $10^{-9}\|\mathbf{x}\|_2$. The 128×128 patch size gives the best performance when fewer than 85% of the measurements are used, slightly better than 32×32, and much better than the rest. Note that these parameters were tuned for time-slice domain reconstruction. For other scenarios, different parameters might be better suited. For BPFA, tuning of the parameters was not necessary due to the adaptability of learning basis. Interested readers can refer to Pilikos and Faul (2017) for further information about initialization.

Hybrid approach with learned dictionaries

Using the RVM with DCT on 128×128 patches has provided the best configuration (Pilikos and Faul, 2016). Nevertheless, the computational time is very long and is orders of magnitude slower than others as we will discuss later on. Operating on 8×8 patches reduces the computational time but simultaneously reduces the reconstruction accuracy (Pilikos and Faul, 2016). In addition, the assumption that DCT provides a sparse representation for every section of a time slice is limiting. Here, we use two hybrid approaches, referred to as RVM-learned and SPGL1-learned, that use more specialized basis per section. We operate on 8×8 patch sizes but using the learned BPFA basis. We use the BPFA basis that is learned for each section as a dictionary for the RVM and for SPGL1 and investigate potential improvements.

TIME-SLICE EXPERIMENTS

By using these algorithmic configurations, we compare against BPFA, POCS, and SPGL1. We ran experiments on 150 sections of time slices extracted from the SEAM-II data set close to the source and far offset (as illustrated in Figure 7), using 8×8 and 128×128 patch sizes. For the experiments on 8×8, we use overlapping patches on the vertical and horizontal directions to avoid edge artifacts in reconstructions. To develop an understanding of the performance, we plot the mean Q with different percentages of receivers used in Figure 8.

We can see that using 8×8 patches, BPFA gives the best performance in general. Then, the RVM using the DCT basis follows in accuracy. After that, SPGL1 performs better from 40% compared to POCS and POCS performs better than SPGL1 below 40%. The same behavior is observed when using a 128×128 patch size. For the RVM with 128×128 patch size, we ran experiments between 20% and 70% and every 10%. This is due to the fact that each run takes significant amount of computation as we will see later on. For POCS and SPGL1, experiments between 5% and 95% every 5% were undertaken. We can see that all algorithms perform much better than their corresponding 8×8 configuration as expected. The RVM performs better than the rest, with SPGL1 better than POCS greater than 35% and POCS better than SPGL1 below that percentage.

The overall trend is the same for 8×8 and 128×128 patches. BPFA on 8×8 patches obtains very similar reconstruction accuracy as the RVM of 128×128 patch size, albeit slightly worse. When we use the learned BPFA basis with the RVM (RVM-learned) and SPGL1 (SPGL1-learned) on 8×8 patches, we can see a great improvement in reconstruction. The RVM with BPFA basis performs much better than its corresponding configuration with DCT on 8×8 patches as well as better than the other algorithms even when they operate on 128×128 patches. The same behavior is exhibited

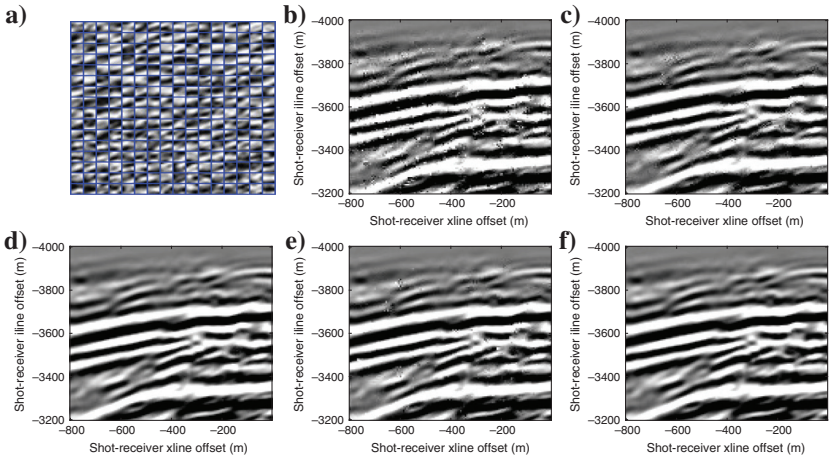

Figure 10. Reconstruction results using various algorithms and dictionaries of basis for the section of a far-offset time slice in Figure 9a. (a) BPFA basis, (b) POCS, 8×8, $Q = 20.12$ db, (c) SPGL1-DCT, 8×8, $Q = 22.28$ db, (d) SPGL1-Learned, 8×8, $Q = 35.40$ db, (e) RVM-DCT, 8×8, $Q = 27.45$ db, and (f) RVM-learned, 8×8, $Q = 38.70$ db.

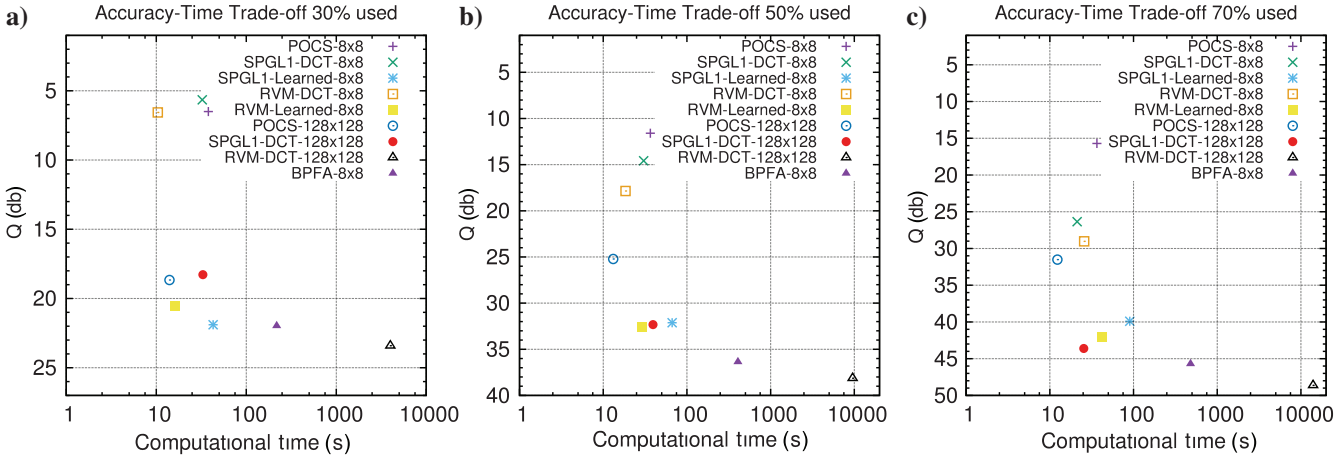

Figure 11. Mean reconstruction accuracy Q against time using various percentages. (a) Using 30% of the receivers, (b) using 50% of the receivers, and (c) using 70% of the receivers.

by SPGL1. Nevertheless, the best is still the RVM with DCT on 128×128 patches but with the added expense of longer run times.

A reconstruction of a section from a time slice is included to visualize the differences between algorithms. Figure 9a shows the original section, and Figure 9b shows the same signal using only 50% of the receivers randomly. The reconstruction of the RVM using DCT with 128×128 patch size is in Figure 9c, the corresponding SPGL1 configuration in Figure 9d and the POCS configuration in Figure 9e. The BPFA reconstruction on the 8×8 size is included in Figure 9f with the learned dictionary of basis in Figure 10a. We can see that BPFA learns the orientation of the signal. Figure 10b shows the reconstruction of POCS with 8×8 patch sizes performing poorly. SPGL1 with the DCT basis on 8×8 patches is shown in Figure 10c, and the corresponding improvement in reconstruction using the learned BPFA basis is illustrated in Figure 10d. We repeat the same reconstruction but using the RVM with DCT on 8×8 patches in Figure 10e, and great improvements are shown using the learned BPFA basis in Figure 10f.

Table 1. Mean Q for the far-offset receiver lines' reconstruction.

Far offset — Reconstruction accuracy in Q [db]			
Decimation rate	30%	50%	70%
RVM-DCT 8×8	14.096	24.522	35.092
RVM-learned 8×8	16.850	34.354	41.279
POCS 8×8	9.917	15.981	21.106
SPGL1-DCT 8×8	9.983	20.143	30.492
SPGL1-learned 8×8	20.084	32.705	37.095
BPFA 8×8	11.545	34.257	42.436
RVM-DCT 128×128	**23.569**	**35.460**	**43.679**
POCS 128×128	15.841	20.882	27.740
SPGL1-DCT 128×128	22.137	31.402	40.975

Trade-off plots of accuracy and time for all algorithms

Obtaining high reconstruction accuracy is essential but is not the only criterion for the usage of an algorithm, with computational time being another. Experiments were performed using the MATLAB packages mentioned earlier, as single-core jobs on machines with an Intel Xeon CPU E5-2650 with 2.00 GHz. The mean time for three different percentages of receivers used (30%, 50%, and 70%) and the accuracy for all algorithms have been recorded over 150 sections of time slices.

Figure 11a shows the results when 30% of the receivers are used. We can clearly see that the configurations of overlapping patches for POCS, RVM with DCT and SPGL1 with DCT on 8×8 patches obtain poor accuracy but are fast. RVM-learned and SPGL1-learned ran fast and at the same time obtain improved reconstruction accuracy. POCS and SPGL1 with DCT on 128×128 obtain better accuracy than the 8×8 configurations of the same algorithms. BPFA on 8×8 obtains the best accuracy out of all the 8×8 configurations, but it is also the slowest of those. RVM with DCT on 128×128 obtains the best accuracy of all, but it is the slowest. When using 50% of the receivers, similar behavior in Figure 11b is obtained. It is worth noting that the RVM-learned on 8×8 patch size obtains better reconstruction accuracy than POCS and SPGL1-DCT on 128×128 and on 8×8 for all. Only BPFA and RVM-DCT on 128×128 are better than RVM-learned. Figure 11c exhibits similar behavior when using 70% of the receivers.

Overall, the best and slowest out of all the algorithms is RVM-DCT on 128×128 patches. Following, BPFA is the best and slowest on 8×8 patches. RVM-learned is in general the second best out of the 8×8 configurations. RVM-learned is orders of magnitude faster than BPFA and RVM-DCT on 128×128 with Q only being worse by a few db. However, note that RVM-learned first requires BPFA to

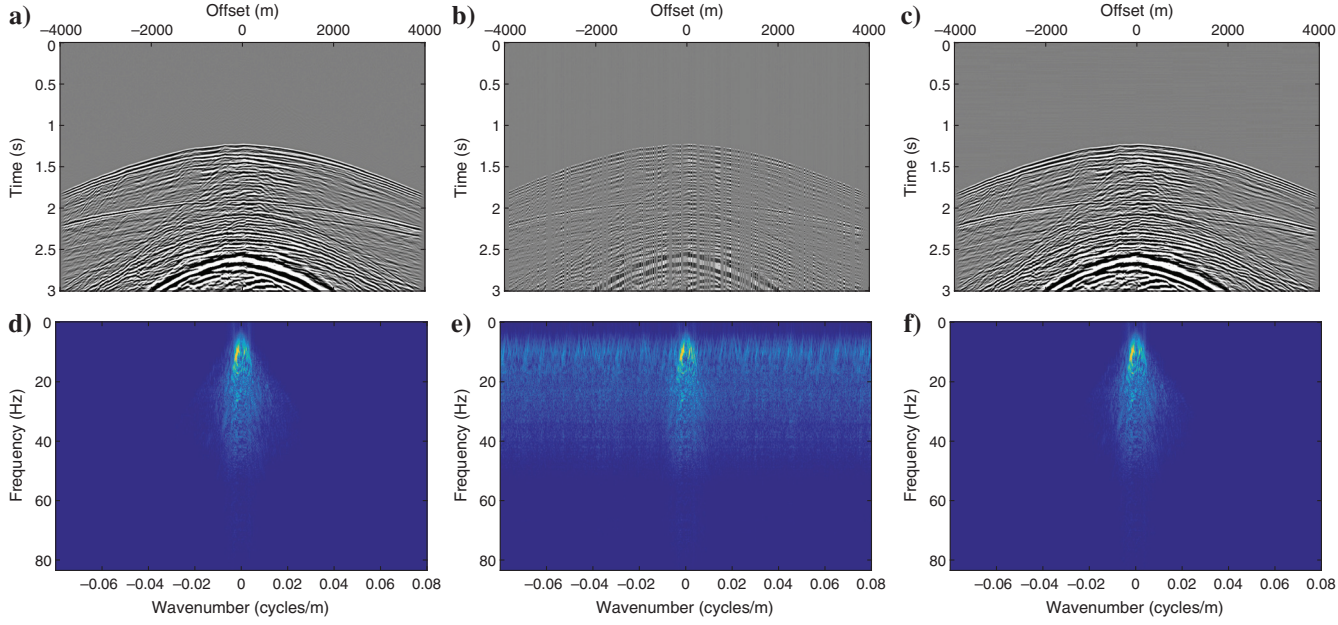

Figure 12. The x-t domain reconstruction for receiver lines (far offset) using the RVM. (a) Original, (b) using 30% of the receivers, (c) RVM-DCT, 128×128, $Q = 28.6$ db, (d) f-k domain of (a), (e) f-k domain of (b), and (f) f-k domain of (c).

DOI:10.1190/geo2019-0200.1

learn the basis functions used. Thus, if these basis functions are not learned offline, RVM-learned would need to wait for BPFA to finish.

x-t DOMAIN EXPERIMENTS

We have shown earlier that the RVM obtains poor reconstruction when operating on data with large gaps. However, to process seismic data in the *x-t* domain entire lines of data points are missing when one receiver is missing. Considering this issue, we decided to operate in the time-slice domain, remove receivers corresponding to data points randomly, reconstruct each section of a time slice, and then re-sort the data in the *x-t* domain. This procedure was followed by Pilikos et al. (2017) as well. We used the 3D synthetic data set generated numerically using the SEAM-II model as input. We extracted 500 time slices, and, from each slice, we extracted two sets of 10 sections of 128×128 (the last section is 128×129). The sections were extracted at the near and far offsets (as shown in Figure 7) to test the reconstruction with different signal structures. To perform irregular undersampling, we created three sampling functions of size 128×1281 with different percentages of receivers kept randomly (30%, 50%, and 70%). These sampling functions were fixed for all time slices to match with entire receivers missing as shown in Figure 4. Experiments in the *x-t* domain regarding the reconstruction accuracy are included for the RVM with the DCT basis, POCS and SPGL1 with the DCT basis on 8×8 and 128×128 patches. Results are also included for BPFA, RVM-learned, and SPGL1-learned on 8×8 patches.

Comparisons regarding reconstruction accuracy

We start the comparisons with far-offset receiver lines. As we have seen in Figure 7, 10 far-offset sections for all time samples are extracted (5000 sections). Then, we use 30%, 50%, and 70% of the receivers and after all reconstructions are finished, we re-sort the data in the *x-t* domain resulting in 128 receiver lines that are far offset. Table 1 summarizes these results where the mean reconstruction accuracy is shown for three decimation rates. We can see that the RVM using DCT on the 128×128 patch size performs better than all other algorithms. From the 8×8 configurations, BPFA is the best, in general, algorithm out of the algorithms with fixed basis. It is also better compared with the algorithms operating on 128×128 patches (namely, POCS and SPGL1-DCT) showing the great reconstruction accuracy that is possible with the learned basis. Using the learned basis from the BPFA with the RVM and SPGL1, we can see improvements in accuracy. RVM-learned improves for all decimation rates when compared with RVM using DCT basis. It even obtains better results than BPFA for a decimation rate of 50%. We can also see similar improvements for SPGL1 when using the learned basis as opposed to fixed DCT. An example of an *x-t* domain reconstruction can be seen in Figure 12.

We will now discuss the reconstruction results for receiver lines that are closer to the source. The data naturally have a more steeply dipping structure. We provide an example of a line of receivers close to the source in Figure 13a along with its respective *f-k* domain in Figure 13c. Figure 13b shows the same signal but using 50% of the receivers with its respective *f-k* domain in Figure 13d. It can be seen that there is incoherent noise in its *f-k* domain, and this should be removed during reconstruction.

Figure 13e shows the reconstruction obtained by the RVM with DCT on a 128×128 patch and using DCT. We can see that at the first time samples, the reconstruction is unstable. This happens at the center of the signal, closest to the source. This is due to the steep dips in the signal and not the lack of data (there is signal in the gray

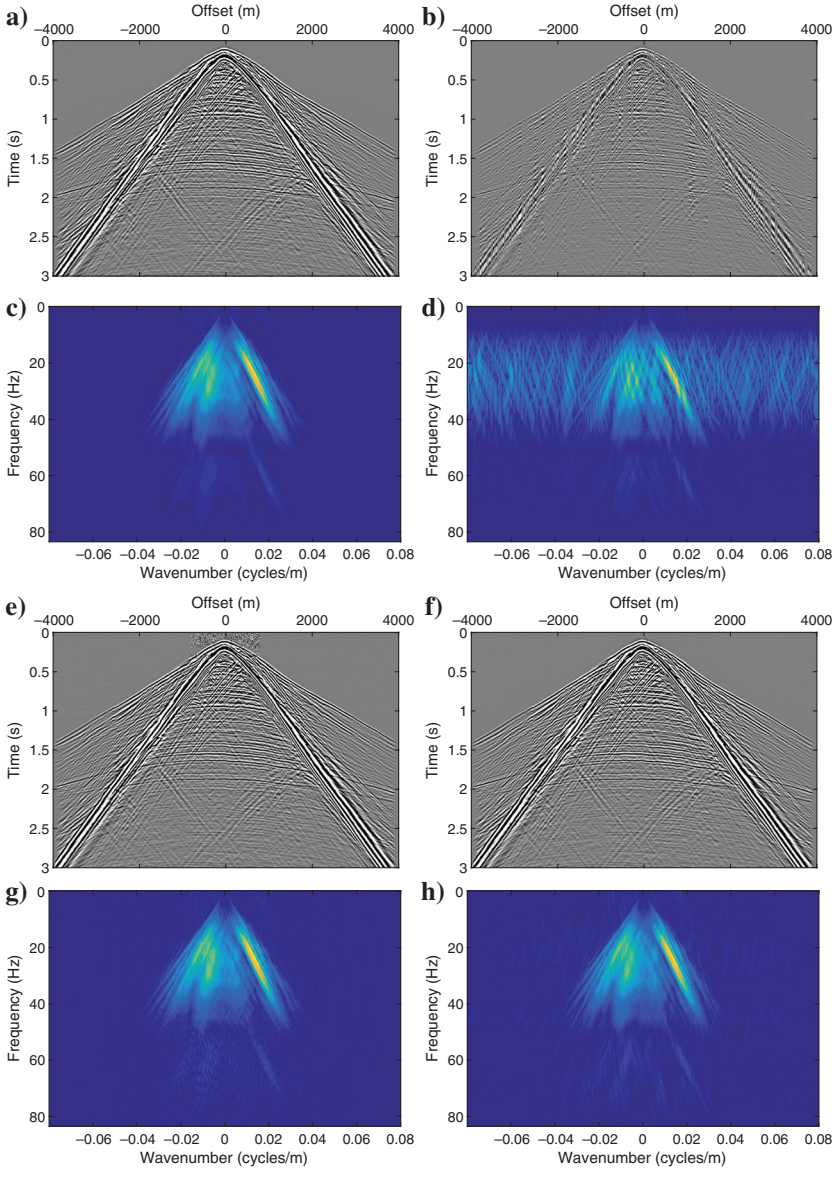

Figure 13. (a) Original, (b) 50% of data, (c) *f-k* domain of (a), (d) *f-k* domain of (b), (e) RVM-DCT, 128×128, $Q = 41.2$ db, (f) RVM-learned, 8×8, $Q = 34.5$ db, (g) *f-k* domain of (e), and (h) *f-k* domain of (f).

areas, albeit a very small amplitude). Essentially, the basis functions used do not contain characteristics with high enough frequencies to capture these changes. Nevertheless, the f-k domain in Figure 13g does not exhibit any aliasing and there is minimal incoherent noise. The seismic signal at the top and center of the x-t domain is very steep, with steeper dips causing problems in reconstruction. This is because we used algorithms on 128×128 patch sizes and the basis functions did not capture the fine details necessary to represent the signal close to the source and at earlier times.

Operating on smaller patch sizes could help in which curved events will appear as linear events (Naghizadeh and Sacchi, 2007). Thus, we repeat the experiment using RVM-learned. The reconstruction can be seen in Figure 13f and its respective f-k domain in Figure 13h. By using the patch size of 8×8, we were able to obtain reconstructions without any distortions around the source location. Nevertheless, the overall reconstruction quality is better when using the 128×128 patch sizes with the exception of the signal close to the source and at earlier times. Thus, we need to separate the seismic signals closer to the source in regions where different algorithms can operate. For example, for the signal close to the source and at earlier times, an algorithm that operates on 8×8 patches should be used and then another algorithm on 128×128 can reconstruct the rest of the regions.

To identify which algorithm works best in each region, we calculate the mean reconstruction accuracy Q over all 128 receiver lines for the three decimation rates in three different regions. We can see that the top center of the signal can be approximately obtained from the first 30 time samples (0.18/0.006 s). Table 2 shows the mean Q over all of the receiver lines for only the first 30 samples. Negative values for Q reflect the poor reconstruction accuracy. The ratio between the norm of the original and the norm of the difference between the original and the reconstruction gives a fraction resulting in a negative exponent (refer to the definition of Q in equation 17). We can see that the RVM using the DCT basis on 8×8 patches gives the best reconstruction accuracy. Using the learned basis from BPFA does not improve the RVM's performance because the basis learned do not capture the characteristics in this region. This is evident from the performance of BPFA as well, which is also poor. The same applies for SPGL1's performance using the learned basis on 8×8 patches. Using the DCT basis by SPGL1 on 8×8 patches gives good results but not as good as the RVM. The POCS results on 8×8 patches are generally poor but still better than the POCS reconstructions on 128×128 because the larger patch size does not help in this region. The same applies to SPGL1 and RVM with DCT on 128×128 with lower Q compared with the respective 8×8 configurations.

We continue the evaluation in other regions as well. The second region is from 31 to 200 time samples, which contains a mixture of steep and curved events. The results are summarized in Table 3. We can see that the RVM using the DCT basis and operating on 128×128 obtains the best reconstruction accuracy in two categories of percentages and SPGL1 using the DCT basis on 128×128 is only slightly better when using 30% of receivers. In general, the configurations operating on 128×128 are now better than those operating in 8×8. In addition, the dictionary of learned basis improves the reconstruction of the RVM and SPGL1 because BPFA is able to learn useful basis. The only exception is when using 30% of receivers. The third region is from 201 to 500 time samples, and the results are summarized in Table 4. The RVM using DCT on 128×128 obtains the best results for two different decimation rates, and BPFA on 8×8

Table 2. Mean Q for the x-t domain for close to source (1–30 time samples).

Close to source (1–30 samples) — Reconstruction accuracy in Q [db]			
Decimation rate	30%	50%	70%
RVM-DCT 8×8	**6.033**	**15.102**	**25.450**
RVM-learned 8×8	0.720	9.009	19.950
POCS 8×8	5.046	8.936	12.662
SPGL1-DCT 8×8	4.832	13.257	23.347
SPGL1-learned 8×8	3.032	12.108	23.828
BPFA 8×8	−7.673	6.153	19.396
RVM-DCT 128×128	−5.936	6.543	17.404
POCS 128×128	−10.005	−1.953	6.109
SPGL1-DCT 128×128	−1.780	6.610	20.992

Table 3. Mean Q for the x-t domain for close to the source (31–200 time samples).

Close to source (31–200 samples) — Reconstruction accuracy in Q [db]			
Decimation rate	30%	50%	70%
RVM-DCT 8×8	4.204	11.080	22.698
RVM-learned 8×8	−0.446	14.535	28.031
POCS 8×8	4.6786	8.3327	12.220
SPGL1-DCT 8×8	3.523	9.772	20.213
SPGL1-learned 8×8	2.315	18.499	28.903
BPFA 8×8	−10.169	19.977	33.752
RVM-DCT 128×128	6.597	**28.475**	**44.550**
POCS 128×128	−1.326	19.315	28.721
SPGL1-DCT 128×128	**6.604**	16.145	33.462

Table 4. Mean Q for the x-t domain for close to source (201–500 time samples).

Close to source (201–500 samples) — Reconstruction accuracy in Q [db]			
Decimation rate	30%	50%	70%
RVM-DCT 8×8	7.695	17.773	28.899
RVM-learned 8×8	6.828	32.634	42.085
POCS 8×8	7.006	11.513	15.904
SPGL1-DCT 8×8	5.803	14.726	24.972
SPGL1-learned 8×8	11.900	28.508	35.667
BPFA 8×8	17.606	**36.921**	44.748
RVM-DCT 128×128	**21.361**	36.017	**45.782**
POCS 128×128	−3.197	23.568	29.128
SPGL1-DCT 128×128	16.825	29.574	39.459

performs better when using 50% of receivers. In addition, the learned basis improve the RVM and SPGL1.

Combining configurations for better accuracy

Using the RVM with DCT on 8×8 at the signal close to the source and at earlier times provides higher reconstruction accuracy with no distortions. For the rest of the regions, the RVM with DCT on 128×128 provides the best accuracy overall. This combination provides the best accuracy and should be used if this is the requirement. If a faster computational time is needed, another combination might be more beneficial.

UNCERTAINTY QUANTIFICATION FOR SOURCE/RECEIVER CONFIGURATIONS

The reconstruction accuracy of algorithms varies depending on the seismic data used. Thus, a metric that quantifies the uncertainty of the model for each prediction would be advantageous. Work toward uncertainty quantification has been undertaken by Pilikos and Philip (2018) and Pilikos and Faul (2019). If the best possible uncertainty quantification is required, BPFA has been shown by Pilikos and Faul (2019) to obtain better results than RVM. Furthermore, the predictive variance of the RVM could be problematic if localized basis functions are used, resulting in the degenerate case (Rasmussen and Quiñonero-Candela, 2005). Here, we use the RVM with DCT to mitigate this as was done by Pilikos and Faul (2019). Ideally, the total uncertainty in the predictions should be small when the reconstruction accuracy is high and vice versa. To quantify the total uncertainty, we use the predictive variance of the RVM obtained per receiver, as defined in equation 16. Then, we take the sum of all the predictive variances for all receivers.

In this section, we illustrate three different configurations of receivers with the corresponding RVM reconstruction, error, and uncertainty map. The configurations vary with respect to the percentage of receivers used (30%, 50%, and 70%). In addition, different seeds are used for random placement of receivers. Figure 14a shows the first configuration using 70% of the receivers. Figure 14b includes the RVM reconstruction with $Q = 44.04$ db. The uncertainty map is included in Figure 14c with the total uncertainty equal to 3.84×10^{-17}. Figure 14d includes the respective reconstruction error. Figure 14e shows the second configuration with 50% of the receivers used. Figure 14f includes the respective RVM reconstruction with $Q = 32.93$ db. The uncertainty map is included in Figure 15a with the total uncertainty equal to 4.73×10^{-17}. The respective reconstruction error is shown in Figure 15b. Finally, the third configuration using 30% of the receivers is shown in Figure 15c. The respective RVM reconstruction is included in Figure 15d with $Q = 24.79$ db. The uncertainty map is shown in Figure 15e with the total uncertainty equal to 5.42×10^{-17}.

These results show that better accuracy corresponds to lower total uncertainty and vice versa. Thus, the predictive variance is able to capture the quality of reconstruction. Further investigation is necessary to understand when the total uncertainty is low enough to enable the respective reconstructions to be suitable for later workflows. Extensions of the RVM's predictive variance could also be further investigated (Rasmussen and Quiñonero-Candela, 2005; Faul and Pilikos, 2016).

FIELD DATA SET EXPERIMENT

We use the Parihaka data set to test the RVM on field data, a 3D seismic image provided by New Zealand Petroleum and Minerals and obtained from the SEG wiki page (SEG, 2018b). All data from the Parihaka data set are prestack time migrated. Figure 16a shows a section from the volume. We removed 30% of the data points

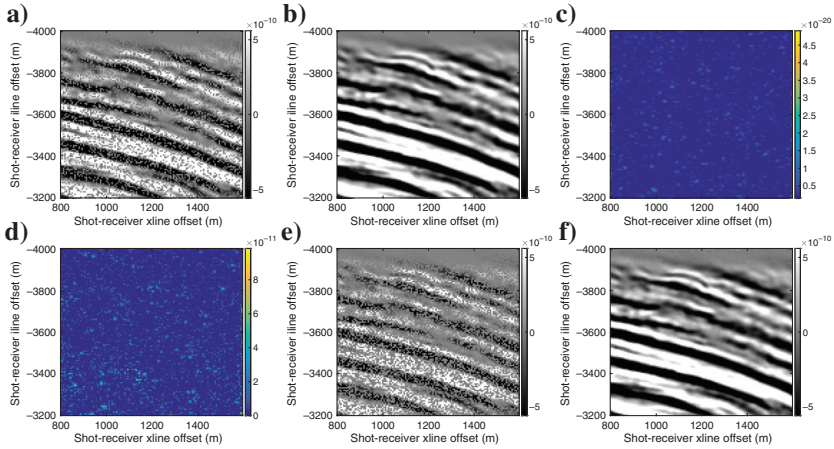

Figure 14. Different source/receiver configurations and uncertainty quantification. (a) First configuration, (b) first RVM reconstruction, $Q = 44.04$ db, (c) first map with uncertainty $= 3.84 \times 10^{-17}$, (d) first reconstruction error, (e) second configuration, and (f) second RVM reconstruction, $Q = 32.93$ db.

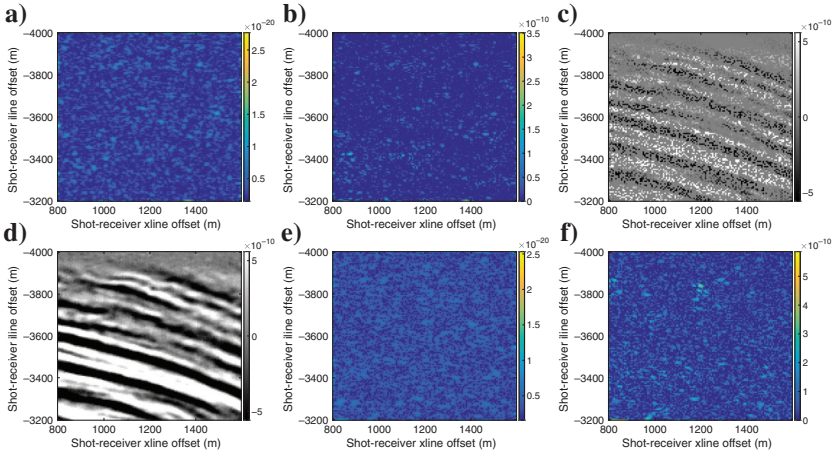

Figure 15. Different source/receiver configurations and uncertainty quantification. (a) Second map, with uncertainty $= 4.73 \times 10^{-17}$, (b) second reconstruction error, (c) third configuration, (d) third RVM reconstruction, $Q = 24.79$ db, (e) third map with uncertainty $= 5.42 \times 10^{-17}$, and (f) third reconstruction error.

randomly as shown in Figure 16b and reconstructed it using the RVM as seen in Figure 16c. The reconstruction error can be seen in Figure 16d with small differences.

DISCUSSION

Using the selected configurations, we compared the RVM on time slices against SPGL1 and POCS. The RVM obtained the best reconstruction accuracy, but it also took the longest to run. Using a patch size of 8×8 improved the computational time, but the reconstruction accuracy was also degraded. To avoid degradation, we used a hybrid approach, referred to as RVM-learned. This algorithm uses the RVM with a dictionary of basis functions learned by another method called BPFA. By exploring all possible spaces, BPFA is able to learn a sparse representation that captures the signal variations of the seismic data. To maintain high reconstruction accuracy and practical computational time, we process 3D seismic data in two dimensions over time resulting in a pseudo 3D seismic data reconstruction. Working directly in higher dimensions such as 3D or 5D (prestack) data is also possible in theory, but the computational time would be orders of magnitude larger.

We experimented with two different types of receiver lines, those that pass close to the source and those at a far offset as in Figure 7. From the experiments of receiver lines that are far offset, the RVM with DCT on 128×128 patches obtained the best reconstruction accuracy for all decimation rates of receivers used. Using RVM-learned on 8×8 patches improved its performance compared to the RVM using DCT on 8×8 patches. For the receiver lines close to the source, the reconstruction is more challenging due to the steeper dips at earlier times. All algorithms that operate on 128×128 patches had problems reconstructing the events at earlier times close to the source due to the very rapid variability. The basis functions used on 128×128 patches did not contain such localized features. However, the algorithms that operated on 8×8 patches reconstructed the region without problems due to the use of smaller basis functions that can capture finer, localized details.

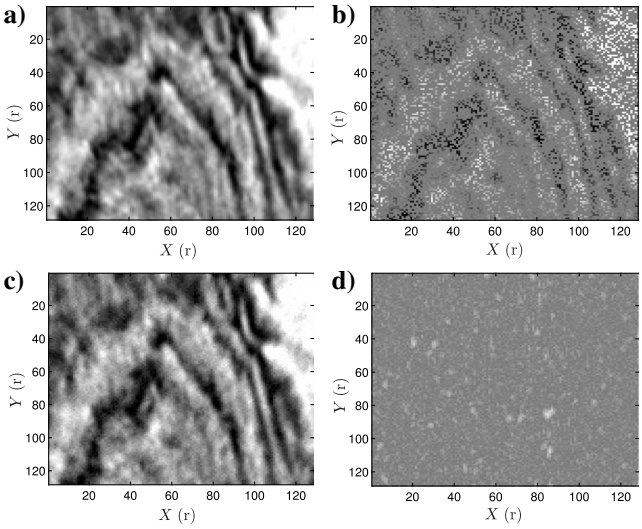

Figure 16. A section from the Parihaka field data set. (a) The original, (b) 30% of data points used, (c) the reconstruction using RVM-DCT, 128×128, $Q = 20.315$ db, and the error in (d).

It has been shown that the reconstruction accuracy of algorithms is different when operating on different regions. Thus, it is essential to obtain a level of confidence for the predictions. We have illustrated that the RVM's predictive variance can be used to quantify uncertainty and investigated source/receiver configurations. When the total uncertainty was low, the reconstruction accuracy was high and vice versa. This could be used for the design of randomized acquisition patterns.

CONCLUSION

Using seismic CS, it is possible to reconstruct undersampled seismic data with high accuracy. A probabilistic data-driven model, called RVM, is suitable for seismic CS because it is composed of a linear combination of basis functions with only a few nonzero coefficients. This results in a sparse model, an assumption necessary for CS to work. By using this model, a predictive distribution is created and used for the prediction of missing receivers as well as uncertainty quantification. In addition, a hybrid approach is used that uses learned dictionaries to improve the reconstruction accuracy. Random sampling and no large gaps in the data are an essential component of this method, with experiments showing that the RVM obtains state-of-the-art data reconstruction without aliasing. Furthermore, it is a useful tool for estimating levels of confidence in the design of a seismic survey by quantifying uncertainty. This uncertainty could be used to evaluate different source/receiver configurations and could guide seismic survey design.

ACKNOWLEDGMENTS

This work was done at the Laboratory for Scientific Computing, Department of Physics, University of Cambridge. I would like to thank BP for their support and for allowing the publication of this research. Special thanks go to A. C. Faul for the great help and suggestions during this work. Furthermore, I would like to thank N. Nikiforakis for the technical input and fruitful discussions. In addition, I thank N. Philip and R. Abma from BP for very useful discussions on seismic data acquisition. I thank SEAM-II for the model and C. Regone of BP for modeling the seismic data as well as New Zealand Petroleum and Minerals for providing the Parihaka field data set. I acknowledge funding from the Engineering and Physical Sciences Research Council through an Industrial CASE studentship (grant no. 1502944).

DATA AND MATERIALS AVAILABILITY

Please contact the author for details.

REFERENCES

Abma, R., 2018, Projection onto convex sets (POCS) software, http://www.freeusp.org/synthetics/POCS_example/, accessed 8 January 2018.

Abma, R., and N. Kabir, 2006, 3D interpolation of irregular data with a POCS algorithm: Geophysics, **71**, no. 6, E91–E97, doi: 10.1190/1.2356088.

Beck, A., and M. Teboulle, 2009, A fast iterative shrinkage-thresholding algorithm for linear inverse problems: SIAM Journal Imaging Sciences, **2**, 183–202, doi: 10.1137/080716542.

Beckouche, S., and J. Ma, 2014, Simultaneous dictionary learning and denoising for seismic data: Geophysics, **79**, no. 3, A27–A31, doi: 10.1190/geo2013-0382.1.

Chen, Y., D. Zhang, W. Huang, S. Zu, Z. Jin, and W. Chen, 2016, Damped rank-reduction method for simultaneous denoising and reconstruction of 5D seismic data: 86th Annual International Meeting, SEG, Expanded Abstracts, 4075–4080, doi: 10.1190/segam2016-13578061.1.

Duijndam, A. J. W., 1988a, Bayesian estimation in seismic inversion — Part 1: Principles: Geophysical Prospecting, **36**, 878–898, doi: 10.1111/j.1365-2478.1988.tb02198.x.

Duijndam, A. J. W., 1988b, Bayesian estimation in seismic inversion — Part 2: Uncertainty analysis: Geophysical Prospecting, **36**, 899–918, doi: 10.1111/j.1365-2478.1988.tb02199.x.

Elad, M., and M. Aharon, 2006, Image denoising via sparse and redundant representations over learned dictionaries: IEEE Transactions on Image Processing, **15**, 3736–3745, doi: 10.1109/TIP.2006.881969.

Faul, A., and G. Pilikos, 2016, The model is simple, until proven otherwise: How to cope in an ever-changing world: Data for Policy, Frontiers of Data Science for Government: Ideas, Practices and Projections.

Fomel, S., and Y. Liu, 2010, Seislet transform and seislet frame: Geophysics, **75**, no. 3, V25–V38, doi: 10.1190/1.3380591.

Gajewski, D., and Y. Xie, 2017, 5-D interpolation with wave-front attributes: Geophysical Journal International, **211**, 897–919, doi: 10.1093/gji/ggx334.

Gan, S., S. Wang, Y. Chen, X. Chen, W. Huang, and H. Chen, 2016, Compressive sensing for seismic data reconstruction via fast projection onto convex sets based on seislet transform: Journal of Applied Geophysics, **130**, 194–208, doi: 10.1016/j.jappgeo.2016.03.033.

Gan, S., S. Wang, Y. Chen, Y. Zhang, and Z. Jin, 2015, Dealiased seismic data interpolation using seislet transform with low-frequency constraint: IEEE Geoscience and Remote Sensing Letters, **12**, 2150–2154, doi: 10.1109/LGRS.2015.2453119.

Gao, J., M. D. Sacchi, and X. Chen, 2013, A fast reduced-rank interpolation method for prestack seismic volumes that depend on four spatial dimensions: Geophysics, **78**, no. 1, V21–V30, doi: 10.1190/geo2012-0038.1.

Gülünay, N., 2003, Seismic trace interpolation in the Fourier transform domain: Geophysics, **68**, 355–369, doi: 10.1190/1.1543221.

Hennenfent, G., and F. J. Herrmann, 2008, Simply denoise: Wavefield reconstruction via jittered undersampling: Geophysics, **73**, no. 3, V19–V28, doi: 10.1190/1.2841038.

Herrmann, F. J., and G. Hennenfent, 2008, Non-parametric seismic data recovery with curvelet frames: Geophysical Journal International, **173**, 233–248, doi: 10.1111/j.1365-246X.2007.03698.x.

Ji, S., Y. Xue, and L. Carin, 2008, Bayesian compressive sensing: IEEE Transactions on Signal Processing, **56**, 2346–2356, doi: 10.1109/TSP.2007.914345.

Jia, Y., and J. Ma, 2017, What can machine learning do for seismic data processing? An interpolation application: Geophysics, **82**, no. 3, V163–V177, doi: 10.1190/geo2016-0300.1.

Jia, Y., S. Yu, and J. Ma, 2018, Intelligent interpolation by Monte Carlo machine learning: Geophysics, **83**, no. 2, V83–V97, doi: 10.1190/geo2017-0294.1.

Jingjie, C., W. Yanfei, and W. Benfeng, 2015, Accelerating seismic interpolation with a gradient projection method based on tight frame property of curvelet: Exploration Geophysics, **46**, 253–260, doi: 10.1071/EG14016.

Kabir, M. N., and D. Verschuur, 1995, Restoration of missing offsets by parabolic Radon transform: Geophysical Prospecting, **43**, 347–368, doi: 10.1111/j.1365-2478.1995.tb00257.x.

Kazemi, N., E. Bongajum, and M. D. Sacchi, 2016, Surface-consistent sparse multichannel blind deconvolution of seismic signals: IEEE Transactions on Geoscience and Remote Sensing, **54**, 3200–3207, doi: 10.1109/TGRS.2015.2513417.

Kazemi, N., and M. D. Sacchi, 2014, Sparse multichannel blind deconvolution: Geophysics, **79**, no. 5, V143–V152, doi: 10.1190/geo2013-0465.1.

Kong, D., and Z. Peng, 2015, Seismic random noise attenuation using shearlet and total generalized variation: Journal of Geophysics and Engineering, **12**, 1024–1035, doi: 10.1088/1742-2132/12/6/1024.

Kreimer, N., and M. D. Sacchi, 2012, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation: Geophysics, **77**, no. 3, V113–V122, doi: 10.1190/geo2011-0399.1.

Kumar, R., C. D. Silva, O. Akalin, A. Y. Aravkin, H. Mansour, B. Recht, and F. J. Herrmann, 2015, Efficient matrix completion for seismic data reconstruction: Geophysics, **80**, no. 5, V97–V114, doi: 10.1190/geo2014-0369.1.

Kutscha, H., and D. Verschuur, 2016, The utilization of the double focal transformation for sparse data representation and data reconstruction: Geophysical Prospecting, **64**, 1498–1515, doi: 10.1111/1365-2478.12362.

Liang, J., J. Ma, and X. Zhang, 2014, Seismic data restoration via data-driven tight frame: Geophysics, **79**, no. 3, V65–V74, doi: 10.1190/geo2013-0252.1.

Liu, B., and M. D. Sacchi, 2004, Minimum weighted norm interpolation of seismic records: Geophysics, **69**, 1560–1568, doi: 10.1190/1.1836829.

Liu, Y., and S. Fomel, 2010, OC-seislet: Seislet transform construction with differential offset continuation: Geophysics, **75**, no. 6, WB235–WB245, doi: 10.1190/1.3479554.

Naghizadeh, M., 2012, Seismic data interpolation and denoising in the frequency-wavenumber domain: Geophysics, **77**, no. 2, V71–V80, doi: 10.1190/geo2011-0172.1.

Naghizadeh, M., and M. D. Sacchi, 2007, Multistep autoregressive reconstruction of seismic records: Geophysics, **72**, no. 6, V111–V118, doi: 10.1190/1.2771685.

Naghizadeh, M., and M. D. Sacchi, 2010a, Beyond alias hierarchical scale curvelet interpolation of regularly and irregularly sampled seismic data: Geophysics, **75**, no. 6, WB189–WB202, doi: 10.1190/1.3509468.

Naghizadeh, M., and M. D. Sacchi, 2010b, On sampling functions and Fourier reconstruction methods: Geophysics, **75**, no. 6, WB137–WB151, doi: 10.1190/1.3503577.

Oristaglio, M., 2012, Seam Phase II land seismic challenges: The Leading Edge, **31**, 264–266, doi: 10.1190/1.3694893.

Oropeza, V., and M. Sacchi, 2011, Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis: Geophysics, **76**, no. 3, V25–V32, doi: 10.1190/1.3552706.

Paisley, J., and L. Carin, 2009, Nonparametric factor analysis with beta process priors: Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 777–784.

Pérez, D. O., D. R. Velis, and M. D. Sacchi, 2013, High-resolution prestack seismic inversion using a hybrid FISTA least-squares strategy: Geophysics, **78**, no. 5, R185–R195, doi: 10.1190/geo2013-0077.1.

Pilikos, G., and A. C. Faul, 2016, Relevance vector machines with uncertainty measure for seismic Bayesian compressive sensing and survey design: 15th IEEE International Conference on Machine Learning and Applications (ICMLA), 925–930.

Pilikos, G., and A. C. Faul, 2017, Bayesian feature learning for seismic compressive sensing and denoising: Geophysics, **82**, no. 6, O91–O104, doi: 10.1190/geo2016-0373.1.

Pilikos, G., and A. C. Faul, 2019, Bayesian modeling for uncertainty quantification in seismic compressive sensing: Geophysics, **84**, no. 2, P15–P25, doi: 10.1190/geo2018-0145.1.

Pilikos, G., A. C. Faul, and N. Philip, 2017, Seismic compressive sensing beyond aliasing using Bayesian feature learning: 87th Annual International Meeting, SEG, Expanded Abstracts, 4328–4332, doi: 10.1190/segam2017-17558742.1.

Pilikos, G., and N. Philip, 2018, Beta process factor analysis for efficient seismic compressive sensing with uncertainty quantification: IEEE 23rd International Conference on Digital Signal Processing (DSP), 1–5.

Rasmussen, C. E., and J. Quiñonero-Candela, 2005, Healing the relevance vector machine through augmentation: Proceedings of the 22nd International Conference on Machine Learning, 689–696.

Sacchi, M., T. Ulrych, and C. Walker, 1998, Interpolation and extrapolation using a high-resolution discrete Fourier transform: IEEE Transactions on Signal Processing, **46**, 31–38, doi: 10.1109/78.651165.

SEG, 2018a, http://seg.org/news-resources/research-and-data/seam, accessed 8 January 2018.

SEG, 2018b, http://wiki.seg.org/wiki/parihaka-3D, accessed 2 February 2018.

Shahidi, R., G. Tang, J. Ma, and F. J. Herrmann, 2013, Application of randomized sampling schemes to curvelet based sparsity promoting seismic data recovery: Geophysical Prospecting, **61**, 973–997, doi: 10.1111/1365-2478.12050.

Siahsar, M. A. N., S. Gholtashi, A. R. Kahoo, W. Chen, and Y. Chen, 2017, Data-driven multitask sparse dictionary learning for noise attenuation of 3D seismic data: Geophysics, **82**, no. 6, V385–V396, doi: 10.1190/geo2017-0084.1.

Stanton, A., N. Kreimer, D. Bonar, M. Naghizadeh, and M. Sacchi, 2012, A comparison of 5D reconstruction methods: 82nd Annual International Meeting, SEG, Expanded Abstracts, doi: 10.1190/segam2012-0269.1.

Stanton, A., M. D. Sacchi, R. Abma, and J. A. Stein, 2015, Mitigating artifacts in projection onto convex sets interpolation: 85th Annual International Meeting, SEG, Expanded Abstracts, 3779–3783, doi: 10.1190/segam2015-5754691.1.

Tipping, M., 2018, Relevance vector machine software, http://www.miketipping.com/downloads.html, accessed 8 January 2018.

Tipping, M. E., 2001, Sparse Bayesian learning and the relevance vector machine: Journal of Machine Learning Research, **1**, 211–244, doi: 10.1162/15324430152748236.

Tipping, M. E., and A. Faul, 2003, Fast marginal likelihood maximisation for sparse Bayesian models: Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics, 3–6.

Trad, D. O., T. J. Ulrych, and M. D. Sacchi, 2002, Accurate interpolation with high resolution time variant Radon transforms: Geophysics, **67**, 644–656, doi: 10.1190/1.1468626.

Trickett, S., L. Burroughs, A. Milton, L. Walton, and R. Dack, 2010, Rank reduction based trace interpolation: 80th Annual International Meeting, SEG, Expanded Abstracts, 3829–3833, doi: 10.1190/1.3513645.

Turquais, P., E. G. Asgedom, W. Sollner, and E. Otnes, 2015, Dictionary learning for signal-to-noise ratio enhancement: 85th Annual International Meeting, SEG, Expanded Abstracts, 4698–4702, doi: 10.1190/segam2015-5846080.1.

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.

van den Berg, E., and M. P. Friedlander, 2007, SPGL1: A solver for large-scale sparse reconstruction software, http://www.cs.ubc.ca/labs/scl/spgl1, accessed 8 January 2018.

van den Berg, E., and M. P. Friedlander, 2009, Probing the Pareto Frontier for basis pursuit solutions: SIAM Journal on Scientific Computing, **31**, 890–912, doi: 10.1137/080714488.

Xu, S., Y. Zhang, D. Pham, and G. Lambaré, 2005, Antileakage Fourier transform for seismic data regularization: Geophysics, **70**, no. 4, V87–V95, doi: 10.1190/1.1993713.

Yu, S., J. Ma, and S. Osher, 2016, Monte Carlo data-driven tight frame for seismic data recovery: Geophysics, **81**, no. 4, V327–V340, doi: 10.1190/geo2015-0343.1.

Yu, S., J. Ma, X. Zhang, and M. D. Sacchi, 2015, Interpolation and denoising of high-dimensional seismic data by learning a tight frame: Geophysics, **80**, no. 5, V119–V132, doi: 10.1190/geo2014-0396.1.

Zhang, D., Y. Zhou, H. Chen, W. Chen, S. Zu, and Y. Chen, 2017, Hybrid rank-sparsity constraint model for simultaneous reconstruction and denoising of 3D seismic data: Geophysics, **82**, no. 5, V351–V367, doi: 10.1190/geo2016-0557.1.

Zhou, M., 2018, Beta process factor analysis software, http://mingyuanzhou.github.io/Code.html, accessed 8 January 2018.

Zhou, M., H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, 2012, Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images: IEEE Transactions on Image Processing, **21**, 130–144, doi: 10.1109/TIP.2011.2160072.

Zhu, L., E. Liu, and J. H. McClellan, 2015, Seismic data denoising through multiscale and sparsity-promoting dictionary learning: Geophysics, **80**, no. 6, WD45–WD57, doi: 10.1190/geo2015-0047.1.

Zwartjes, P. M., and M. D. Sacchi, 2007, Fourier reconstruction of nonuniformly sampled, aliased seismic data: Geophysics, **72**, no. 1, V21–V32, doi: 10.1190/1.2399442.