Convolutional Sparse Coding for Noise Attenuation in Seismic Data

Zhaolun Liu¹ and Kai Lu²

¹ *Formerly King Abdullah University of Science and Technology, Department of Earth
Science and Engineering, Thuwal, Saudi Arabia; presently Princeton University,
Department of Geosciences, Princeton, NJ 08544, USA. E-mail: zhaolunl@princeton.edu;*

² *Formerly King Abdullah University of Science and Technology, Department of Earth
Science and Engineering, Thuwal, Saudi Arabia; Kai.lu@kaust.edu.sa.*

(October 20, 2020)

Running head: **CSC Denoising**

ABSTRACT

We propose convolutional sparse coding (CSC) to attenuate noise in seismic data. CSC gives a data-driven set of basis functions whose coefficients form a sparse distribution. The noise attenuation method by CSC can be divided into the training and the denoising phases. Seismic data with a relatively high signal-to-noise ratio are chosen for training to get the learned basis functions. Then, we use all (or a subset) of the basis functions to attenuate the random or coherent noise in the seismic data. Numerical experiments on synthetic data show that CSC can learn a set of shifted invariant filters, which can reduce the redundancy of learned filters in the traditional sparse-coding denoising method. CSC achieves good denoising performance when training by the noisy data and better performance when training on a similar but noiseless data set. The numerical results from the field data test indicate that CSC can effectively suppress seismic noise in the complex

field data. By excluding filters with coherent noise features, the proposed method can further attenuate the coherent noise and separate ground roll.

INTRODUCTION

Seismic data may suffer from different sources of noise during field acquisition, and this noise generally falls into two categories: (1) coherent noise such as ground roll and multiples; (2) incoherent (random) noise generated by activities in the environment such as the acquisition truck, drill rig, seismic vessels, wind and animal movements. Seismic noise attenuation is one of the critical components of seismic data processing. This paper proposes a convolutional type of sparse-coding based denoising method for the attenuation of coherent and random noise.

Traditional seismic denoise methods exploit the different characteristics of the noise and the seismic data in a transform domain, in which the signal and noise have different characteristics. For example, the bandpass, $f - k$, and $k_x - k_y$ filtering method will separate the undesirable noise from the signal in the Fourier domain (Yilmaz, 2001). Chanerley and Alexander (2002) use the stationary wavelet transform as an alternative to bandpass filtering for denoising. The curvelet transform is used by Hennenfent and Herrmann (2006) to attenuate both random and coherent noise in seismic data. Ibrahim and Sacchi (2014) use the hyperbolic Radon transform for deblending seismic data obtained from a survey with blended sources. The seislet transform is used for seismic denoising by Chen (2016). Other seismic random noise attenuation methods include the nonlocal means method (Bonar and Sacchi, 2012) and the sparsity and low-rank regularization method (Li et al., 2020). Coherent noise attenuation methods include prediction filters (Guitton, 2002), empirical mode decomposition (Bekara and Van der Baan, 2009), and nonstationary polynomial fitting (Liu et al., 2011).

The analytic transform mentioned above is a model-driven process based on a formulated

mathematical model of the data. The performance may not be satisfactory due to the inability to adapt to changing data structures (Zhu et al., 2015). Alternatively, the sparse dictionary learning method is a data-driven process, which learns its dictionary so that complicated data can be represented by a sparse set of weighted basis functions. Kaplan et al. (2009) use the sparse-coding algorithm to attenuate both coherent and incoherent noise. Beckouche and Ma (2014) evaluate a dictionary learning method for seismic-data denoising. Zhu et al. (2015) develop a seismic data denoising method based on a parametric dictionary learning scheme. Chen et al. (2016) combine learning-based dictionaries and fixed-basis transforms and propose a double-sparsity dictionary to better handle the special features of seismic data. Li et al. (2018) propose a novel method based on unsupervised learning to recover the weak residual signal from seismic data. Their synthetic and real data examples indicate that this method is capable of recovering signal even if it is orders of magnitude weaker than the surrounding noise.

Most of the sparse coding (SC) denoising methods partition the seismic data into overlapping patches, and process each patch separately. These methods, however, ignore the consistency of structures in overlapping patches, for which the learned features often contain shifted versions of the same features so that latent structures of the underlying signal may be lost when dividing it into small patches (Bristow et al., 2013; Heide et al., 2015). In this paper, we propose a convolutional sparse coding (CSC) based seismic denoising method to address the consistency issue. As opposed to SC, CSC operates on whole seismic images, thereby seamlessly capturing the correlation between local neighborhoods.

The CSC method can learn a set of translationally invariant dictionaries for a large variety of different training signals (Xiong et al., 2019). It has been demonstrated effective for solving problems in neural and brain information processing (Jas et al., 2017; Peter et al.,

2017), and in different image processing tasks, such as image inpainting (Heide et al., 2015), super-resolution (Gu et al., 2015), high dynamic range imaging (Serrano et al., 2016), and high-dimensional signal reconstructions (Choudhury et al., 2017). CSC is shown to be tightly connected to convolutional neural networks (CNN) by (Papayan et al., 2017a) and has been applied in the seismic least-squares migration problem (Liu and Schuster, 2019; Liu et al., 2020). Al-Madani et al. (2019) apply CSC for seismic random noise attenuation and interpolation. They didn't talk about how to use CSC to attenuate the coherent noise, and the efficacy of this method is validated with synthetic data. However, further studies are needed to assess its capabilities.

The CSC problem can be solved by using the Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011) in the Fourier domain (Wohlberg, 2016; Heide et al., 2015) or in the space domain (Papayan et al., 2017b; Zisselman et al., 2019), which alternate between finding the dictionary (dictionary learning problem) and then finding the sparse coefficients (sparse pursuit problem). In this paper, we will apply the fast and flexible CSC method proposed by Heide et al. (2015) to the seismic denoising process.

We first introduce the theory of the fast and flexible CSC method and propose a workflow of noise attenuation by CSC. Then we apply the CSC denoising method to synthetic data and field records. Numerical results indicate that the proposed method achieves good denoising performance.

THEORY

The convolutional sparse coding problem can be defined as finding the optimal $\mathbf{d}$ and $\mathbf{z}$ that minimize the following objective function (Heide et al., 2015):

$$\arg \min_{\mathbf{d}, \mathbf{z}} \frac{1}{2} \|\mathbf{x} - \mathbf{M} \sum_{k=1}^K \mathbf{d}_k * \mathbf{z}_k\|_2^2 + \beta \sum_{k=1}^K \|\mathbf{z}_k\|_1 + \sum_{k=1}^K \text{ind}_C(\mathbf{d}_k), \quad (1)$$

where $\mathbf{x}$ is an $m \times n$ image in vector form, $\mathbf{d}_k$ refers to the k -th $d \times d$ filter in vector form, $\mathbf{z}_k$ is a vector of sparse coefficients with size $(m + d - 1) \times (n + d - 1)$, β controls the l_1 penalty, and $*$ denotes the 2D convolution operator. $\mathbf{M}$ is a binary diagonal matrix that masks out the boundaries of the padded estimation $\sum_{k=1}^K \mathbf{d}_k * \mathbf{z}_k$. The term $\text{ind}_C(\cdot)$ is an indicator function:

$$\text{ind}_C(\mathbf{d}) = \begin{cases} 0 & \mathbf{d} \in C \\ 1 & \mathbf{d} \notin C, \end{cases} \quad (2)$$

which is defined on the convex set of the constraints $C = \{\mathbf{d} \mid \|\mathbf{d}\|_2^2 \leq 1\}$. Equation 1 can be expressed as the following sum of functions

$$\arg \min_{\mathbf{d}, \mathbf{z}} f_1(\mathbf{D}\mathbf{z}) + \sum_{k=1}^K (f_2(\mathbf{z}_k) + f_3(\mathbf{d}_k)), \quad (3)$$

where,

$$f_1(\mathbf{v}) = \frac{1}{2} \|\mathbf{x} - \mathbf{M}\mathbf{v}\|_2^2, \quad f_2(\mathbf{v}) = \beta \|\mathbf{z}_k\|_1, \quad f_3(\mathbf{v}) = \text{ind}_C(\mathbf{v}). \quad (4)$$

Here, $\mathbf{z} = [\mathbf{z}_1^T, \dots, \mathbf{z}_K^T]^T$ is the coefficient matrix and $\mathbf{D} = [\mathbf{D}_1, \dots, \mathbf{D}_K]$ is a concatenation of Toeplitz matrices, each one representing a convolution with respect to the filter $\mathbf{d}_k$. Equation 3 is a sum of functions f_i , which are simple to optimize individually. However, computing their sum is challenging. Following Heide et al. (2015), equation 3 is a bi-convex problem for $\mathbf{z}$ (or $\mathbf{d}$) when $\mathbf{d}$ (or $\mathbf{z}$) is fixed. So, we can use the alternating coordinate descent method to solve it. First, compute the filter update:

$$\arg \min_{\mathbf{d}} f_1(\mathbf{Z}\mathbf{d}) + \sum_{k=1}^K f_3(\mathbf{d}_k). \quad (5)$$

where $\mathbf{Z} = [\mathbf{Z}_1, \dots, \mathbf{Z}_K]$ is a concatenation of Toeplitz matrices, each one representing a convolution with respect to the coefficient $\mathbf{d}_k$ and $\mathbf{d} = [\mathbf{d}_1^T, \dots, \mathbf{d}_K^T]^T$. Then do the coefficient update:

$$\arg \min_{\mathbf{z}} f_1(\mathbf{D}\mathbf{z}) + \sum_{k=1}^K f_2(\mathbf{z}_k). \quad (6)$$

Repeat the above two steps until there is no more progress in both directions.

Generalization of the Objective Function

The above two subproblems have the same format and so we can define their generalized form as

$$f(\mathbf{K}\mathbf{u}) = \sum_{i=1}^I f_i(\mathbf{K}_i\mathbf{u}), \quad (7)$$

where $\mathbf{K} = [\mathbf{K}_1, \mathbf{K}_2, \dots, \mathbf{K}_I]^T$. $\mathbf{K}_i$ are arbitrary matrices, and I is the number of functions. Here, $\mathbf{u}$ can be $\mathbf{d}$ or $\mathbf{z}$. For example, if $\mathbf{K}_1 = \mathbf{D}$, $\mathbf{K}_2\mathbf{u} = \mathbf{z}_1$, $\mathbf{K}_3\mathbf{u} = \mathbf{z}_2$, $\dots$, $\mathbf{K}_i\mathbf{u} = \mathbf{z}_{i-1}$, $\dots$, $\mathbf{K}_{K+1}\mathbf{u} = \mathbf{z}_K$ and $f_2 = f_3 = \dots = f_{K+1}$, the problem for the coefficient update can be written as:

$$\arg \min_{\mathbf{u}} f(\mathbf{K}\mathbf{u}) = \arg \min_{\mathbf{z}} f_1(\mathbf{D}\mathbf{z}) + \sum_{k=1}^K f_2(\mathbf{z}_k). \quad (8)$$

Heide et al. (2015) use the alternating direction method of multipliers (ADMM) (Boyd et al., 2011) to solve for $\mathbf{u}$ in equation 8. We will see that the resulting minimization by ADMM becomes separable for all the f_i . For example, the following problem is the same as the problem in equation 8:

$$\arg \min_{\mathbf{u}} h(\mathbf{u}) + f(\mathbf{y}) \quad \text{subject to} \quad \mathbf{K}\mathbf{u} = \mathbf{y}, \quad (9)$$

where $h(\mathbf{u}) = 0$. Its augmented Lagrangian can be written as:

$$L_\rho(\mathbf{u}, \mathbf{y}, \boldsymbol{\alpha}) = h(\mathbf{u}) + f(\mathbf{y}) + \boldsymbol{\alpha}^T(\mathbf{K}\mathbf{u} - \mathbf{y}) + (\rho/2)\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2. \quad (10)$$

The scaled form of the augmented Lagrangian is:

$$L_\rho(\mathbf{u}, \mathbf{y}, \boldsymbol{\alpha}) = h(\mathbf{u}) + f(\mathbf{y}) + (\rho/2)\|\mathbf{K}\mathbf{u} - \mathbf{y} + \boldsymbol{\lambda}\|_2^2, \quad (11)$$

where $\boldsymbol{\lambda} = (1/\rho)\boldsymbol{\alpha}$. The ADMM of equation 11 is described by the following three steps.

- $\mathbf{u}$ update:

$$\begin{aligned} \mathbf{u}^{k+1} &= \arg \min_{\mathbf{u}} L_\rho(\mathbf{u}^k, \mathbf{y}^k, \boldsymbol{\lambda}^k), \\ &= \arg \min_{\mathbf{u}} \|\mathbf{K}\mathbf{u}^k - \mathbf{y}^k + \boldsymbol{\lambda}^k\|_2^2. \end{aligned} \quad (12)$$

- $\mathbf{y}$ update:

$$\begin{aligned} \mathbf{y}^{k+1} &= \arg \min_{\mathbf{y}} L_\rho(\mathbf{u}^{k+1}, \mathbf{y}^k, \boldsymbol{\lambda}^k), \\ &= \arg \min_{\mathbf{y}} \rho \left\{ f(\mathbf{y}^k)/\rho + \frac{1}{2}\|\mathbf{K}\mathbf{u}^{k+1} - \mathbf{y}^k + \boldsymbol{\lambda}^k\|_2^2 \right\}. \end{aligned} \quad (13)$$

- $\boldsymbol{\lambda}$ update:

$$\begin{aligned} \boldsymbol{\lambda}^{k+1} &= \arg \max_{\boldsymbol{\lambda}} L_\rho(\mathbf{u}^{k+1}, \mathbf{y}^{k+1}, \boldsymbol{\lambda}^k), \\ &= \boldsymbol{\lambda}^k + (\mathbf{K}\mathbf{u}^{k+1} - \mathbf{y}^{k+1}). \end{aligned} \quad (14)$$

Because f is a sum of f_i , equation 13 can be separated as (Parikh and Boyd, 2014):

$$\begin{aligned} \mathbf{y}_i^{k+1} &= \arg \min_{\mathbf{y}_i} f_i(\mathbf{y}_i^k)/\rho + \frac{1}{2}\|\mathbf{K}_i\mathbf{u}_i^{k+1} - \mathbf{y}_i^k + \boldsymbol{\lambda}_i^k\|_2^2, \\ &\text{for all } i \in \{1, \dots, I\}. \end{aligned} \quad (15)$$

Appendix A shows how to solve the problems defined by equations 12 and 15.

Workflow

[Figure 1 about here.]

The workflow of CSC for noise attenuation is shown in Figure 1. It includes the following two steps:

- Training phase: solve for the filter $\mathbf{d}_k$ and the coefficient vector $\mathbf{z}_k$. The training data are chosen with a relatively high signal-to-noise ratio (SNR). We remark that training on high-SNR data is not necessary because the CSC process has a noise rejection capability like the traditional SC (Elad and Aharon, 2006). However, training on high-SNR data will have better denoising performance than training on the noisy data because it will learn cleaner filters. The filter size should be larger than one wavelength.
- Denoising phase: solve for the coefficient vector $\mathbf{z}_k$ with the knowledge of the learned vector $\mathbf{d}_k$. Change the value of the coefficient β for the l_1 penalty function to control the degree of noise removal. For coherent noise, remove some of the learned filters that are more indicative of noise than signal, and the remaining filters are used for denoising.

NUMERICAL EXAMPLES

The performance of CSC for denoising seismic data is demonstrated with two examples: (1) synthetic data from the Marmousi model, and (2) field data from Saudi Arabia. The synthetic data example tests the performance of CSC in removing random noise in seismic data, and the field data example demonstrates the removal of coherent noise. The SNR used in our synthetic test is defined as follows:

$$\text{SNR} = 10 \log_{10} \left(\frac{\|X_{\text{signal}}\|_2^2}{\|X_{\text{noise}}\|_2^2} \right), \quad (16)$$

where X_{signal} and X_{noise} denote the signal and noise, respectively.

[Figure 2 about here.]

Synthetic Test

The first example is a 2-D synthetic dataset calculated from the Marmousi model, as shown in Figure 2. There are 116 sources with a 40 m interval and 231 receivers with a 20 m interval located along the free surface at the top of the model. We use a 15 Hz Ricker wavelet source, with a 4 s recording time at 1 ms sampling. For each common shot gather (CSG), we extract part of data for test, where t is from 1.6s to 2.5s, and the maximum source-receiver offset is 2 km. For example, the extracted data from the second CSG are displayed in Figure 3a, and the size is 181×100 .

We first compare the denoising performance of the CSC method versus the traditional SC method based on the K-SVD algorithm proposed by Elad and Aharon (2006). We use the noisy data with a SNR of 4.27 (Figure 3b) to train the dictionaries. We use 32 11×11 filters for CSC, which are initialized randomly. For the patch-based traditional SC method, we first divide the noisy data set into overlapping patches each with size 11×11 and the stride size is 1. The total number of the patches is 171×90 , and we randomly select 10,000 as a training set. The number of the atoms (filters) is set to the size of the filters, which makes a dictionary of 121 atoms. A factor of redundancy of two is also a popular choice (Beckouche and Ma, 2014) but here we set the factor of redundancy as one. The atoms are initialized randomly, and the atom sparsity level is set to ten.

Figure 4a and 4b show the learned filters by the convolutional and traditional SC methods. If we compare the filters marked by the red rectangles in Figure 4, we see that the traditional SC method learns shifted versions of the same features. The filters are

learned from the red area of the data shown in Figure 3a. The denoised results by the convolutional and traditional SC methods are shown in Figure 3c and 3d, respectively. The SNRs of the denoised data by the convolutional and traditional SC methods have increased to 14.13 and 13.95, respectively. From the comparison, we see that CSC denoised results may have better SNR than that from the traditional sparse coding method.

[Figure 3 about here.]

[Figure 4 about here.]

[Figure 5 about here.]

[Figure 6 about here.]

[Figure 7 about here.]

We then test the performance of denoising by CSC by training the filters from a clean data set. Five CSGs with high SNR are selected for training. There are 50 learned filters (see Figure 6) and the filter size is 55×55 . Assume that the second CSG is contaminated with some noise (see Figure 5b) and its SNR is -1.76. The denoised CSG is shown in Figure 5c and its SNR increases to 14.5. We can change the coefficient β of the l_1 penalty function to control the noise level. Figures 7a and 7b show the denoised results by setting β to 0.2 and 5, respectively. We observe that more noise is removed with larger values of β . But it may also hurt some useful signals, which can be seen in the comparison of the residuals in Figures 7c and 7d.

[Figure 8 about here.]

[Figure 9 about here.]

[Figure 10 about here.]

[Figure 11 about here.]

Field Data Test

Seismic data are recorded over the Qademah fault system (Li et al., 2019; Liu et al., 2019), approximately 30 km north of the KAUST campus, Saudi Arabia. There are 228 sources with a source spacing of 10 m. Each CSG has 456 receivers with a receiver spacing of 5 m. The dataset is first bandpass filtered with a frequency range from 10 Hz to 60 Hz. Ten CSGs are chosen for training, and one of CSGs is shown in Figure 8a. Its offset range is from 0.2 km to 1.3 km. The surface waves are muted out in the training set. There are 30 21×21 learned filters displayed in Figure 8b. Some features of the coherent noise are learned from the training data set, as indicated by the red boxes in Figure 8b.

We select the first CSG (see Figure 9a) for denoising. Its offset range is from 0.8 km to 1.9 km. β is set to 0.8 during the denoising phase. Using all the learned filters in Figure 8b for denoising gives the denoised result displayed in Figure 9b. Its corresponding residual is shown in Figure 10a. There is still coherent noises in the area indicated by the red box in Figure 9b. The reason is that the learned filters include features with coherent noise.

Next we exclude the filters with noise features indicated by the red boxes in Figure 8b. Then, we apply the remaining filters for denoising. The denoised result and its residual are shown in Figures 9c and Figure 10b, respectively. We see that the noise level within the red box of Figure 9b is reduced by only using the selected filters. We continue to exclude

more filters as indicated by the yellow boxes in Figure 8b during the denoising phase. The denoised result is shown in Figure 9d and its residual is shown in Figure 10c. From the residuals in Figure 10, we see that more coherent noise is removed.

Next, we choose the near-offset traces with offset ranging from 0.05 km to 0.35 km (see Figure 11a) contaminated with surface waves to perform the test. We use all the filters shown in Figure 8b for denoising. The denoised data and the residual are shown in Figures 11b and 11c, respectively. The learned filters do not include the features from the surface waves, which are therefore removed from the data set.

DISCUSSION

In the synthetic data example, for simplicity, we test our denoising algorithm on only a part of the CSG. The generalization for the whole CSG is straightforward. From the synthetic results, we see that for CSC denoising, there are two training options: using the noisy data set itself, or training on a similar but noiseless data set. By comparing the learned filters from the convolutional and traditional SC methods, we find that the filters from the traditional SC method are highly redundant because they have to capture all shifted copies of the filters.

The computational cost of CSC is discussed in Papyan et al. (2017b). Papyan et al. (2017b) show that the dictionary learning step in CSC can be solved in the spatial domain by K-SVD instead of ADMM in the Fourier domain proposed by Heide et al. (2015), the method used in this paper. They conclude that the K-SVD-based method scales linearly with the global dimension, while the method used in this paper grows as $N \log(N)$ where N is the dimension of signal $\mathbf{x}$. So, when the dimension of data is larger, we suggest using

KSVD-based CSC algorithm (Papyan et al., 2017b; Zisselman et al., 2019; Xiong et al., 2019).

The CSC-based denoising method is an unsupervised learning process that differs from supervised learning processes like the CNN-based denoising method suggested by Yu et al. (2019). It does not heavily depend on the availability of a huge amount of training data. However, it may need human intervention for inspecting the quality of the filters, and we may need to visually separate the noisy filters from the clean filters.

CONCLUSION

We propose a convolutional sparse coding (CSC) based seismic denoising method to address the consistency issue of sparse coding (SC). We develop a procedure that can attenuate the coherent and random noise in seismic data: the seismic data with a relatively high signal-to-noise ratio are chosen for training to get the learned basis functions, and then we use all (or a subset) of the basis functions to attenuate the random or coherent noise in the seismic data. For random noise, the value of the coefficient of the L-1 penalty function is selected by a trial-and-error procedure for denoising. For the coherent noise attenuation, the learned basis functions must exclude coherent noise features, such as ground roll.

We show that CSC can learn a set of shifted invariant filters, which can reduce the redundancy of learned filters in the traditional SC denoising method. The synthetic numerical tests indicate that CSC can achieve better performance when training on a similar but noiseless data set compared to train on a noisy data set. Even training with noisy data, CSC still has good denoising performance. The numerical results from the field data test indicate that CSC can effectively suppress seismic noise in the complex field data. By

excluding filters with features of coherent noise, the proposed method can further attenuate the coherent noise and separate ground roll. When attenuating the coherent noise, it needs human intervention to inspect the quality of the filters. The limitation may be released by labeling the basis function according to a physical attribute that differentiates signal from noise, which will be our next study.

ACKNOWLEDGMENTS

The research reported in this publication was supported by the King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia. We are grateful to the sponsors of the Center for Subsurface Imaging and Modeling Consortium for their financial support. For computer time, this research used the resources of the Supercomputing Laboratory at KAUST and the IT Research Computing Group. We thank them for providing the computational resources required for carrying out this work.

APPENDIX A

SUBPROBLEM OF ADMM

We will show how to solve the subproblems defined by equations 12 and 15 in the ADMM procedure. For the quadratic problem in equation 12, its solution is

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}} \|\mathbf{K}\mathbf{u} - \boldsymbol{\tau}\|_2^2 = (\mathbf{K}^T \mathbf{K})^{-1} (\mathbf{K}^T \boldsymbol{\tau}), \quad (\text{A-1})$$

where $\boldsymbol{\tau} = \mathbf{y} - \boldsymbol{\lambda}$.

For the problem defined in equation 15, it can be divided into the following three problems.

1, when $f_i = f_1(\mathbf{v}) = \frac{1}{2}\|\mathbf{x} - \mathbf{M}\mathbf{v}\|_2^2$, the problem becomes

$$\mathbf{y}_{1opt} = \arg \min_{\mathbf{y}_1} \frac{1}{2\rho}\|\mathbf{x} - \mathbf{M}\mathbf{y}_1\|_2^2 + \frac{1}{2}\|\mathbf{K}_1\mathbf{u}_1 - \mathbf{y}_1 + \boldsymbol{\lambda}_1\|_2^2, \quad (\text{A-2})$$

and its solution is

$$\mathbf{y}_{1opt} = (\mathbf{I} + \frac{1}{\rho}\mathbf{M}^T\mathbf{M})^{-1}(\frac{1}{\rho}\mathbf{M}^T\mathbf{x} + \mathbf{K}_1\mathbf{u}_1 + \boldsymbol{\lambda}_1). \quad (\text{A-3})$$

2, when $f_i = f_2(\mathbf{v}) = \beta\|\mathbf{v}\|_1$, the problem becomes:

$$\mathbf{y}_{2opt} = \arg \min_{\mathbf{y}_2} \frac{\beta}{\rho}\|\mathbf{y}_2\|_1 + \frac{1}{2}\|\mathbf{K}_2\mathbf{u}_2 - \mathbf{y}_2 + \boldsymbol{\lambda}_2\|_2^2. \quad (\text{A-4})$$

This is a soft threshold problem. Assume that $\mathbf{v} = \mathbf{K}_2\mathbf{u}_2 + \boldsymbol{\lambda}_2 = (v_1, v_2, \dots)^T$, $\mathbf{y}_2 = (y^1, y^2, \dots)^T$, and its solution is:

$$\mathbf{y}_{2opt}^i = \begin{cases} 0 & \frac{\beta}{\rho} \leq v_i \leq \frac{\beta}{\rho}, \\ v_i - \frac{\beta}{\rho} & v_i > \frac{\beta}{\rho}, \\ v_i + \frac{\beta}{\rho} & v_i < \frac{\beta}{\rho}, \end{cases} \quad (\text{A-5})$$

The above solution can be expressed in a vector form:

$$\mathbf{y}_{2opt} = \max(\mathbf{1} - \frac{\beta}{\rho} \frac{\mathbf{1}}{|\mathbf{v}|}, \mathbf{0}) \circ \mathbf{v},$$

where $\circ$ is the element-wise product operator, $1/|\mathbf{v}|$ is a vector where each of its elements is the inverse of the absolute value of the corresponding element in vector $\mathbf{v}$. $\mathbf{1}$ and $\mathbf{0}$ are the all-one and all-zero vectors, respectively. $\max(\mathbf{a}, \mathbf{b})$ will returns an array with the largest elements taken from vector $\mathbf{a}$ or $\mathbf{b}$.

3, when $f_i = f_3(v) = \text{ind}_C(\mathbf{v})$ where the convex set C is $\{\mathbf{d} \mid \|\mathbf{d}\|_2^2 \leq 1\}$, assume $\mathbf{b} = \mathbf{K}_3\mathbf{u}_3 + \boldsymbol{\lambda}_3$ and the problem becomes:

$$\mathbf{y}_{3opt} = \arg \min_{\mathbf{y}_3 \in C} \frac{1}{2}\|\mathbf{y}_3 - \mathbf{b}\|_2^2 \quad (\text{A-6})$$

if $\mathbf{b} \in C$, $\mathbf{y}_{3opt} = \mathbf{b}$. if $\mathbf{b} \notin C$, $\mathbf{y}_{3opt}$ will have the minimum distance to $\mathbf{b}$. So, $\mathbf{y}_{3opt} = \frac{\mathbf{b}}{\|\mathbf{b}\|_2}$, as shown in Figure 12.

[Figure 12 about here.]

If we define the proximal operator as:

$$\text{prox}_f(v) = \arg \min_x \left(f(x) + \frac{1}{2} \|x - v\|_2^2 \right), \quad (\text{A-7})$$

the above three solutions can be expressed as:

$$\begin{aligned} \text{prox}_{\frac{f_1}{\rho}}(\mathbf{K}_1 \mathbf{u}_1 + \boldsymbol{\lambda}_1) &= (\mathbf{I} + \frac{1}{\rho} \mathbf{M}^T \mathbf{M})^{-1} (\frac{1}{\rho} \mathbf{M}^T \mathbf{x} + \mathbf{K}_1 \mathbf{u}_1 + \boldsymbol{\lambda}_1), \\ \text{prox}_{\frac{f_2}{\rho}}(\mathbf{K}_2 \mathbf{u}_2 + \boldsymbol{\lambda}_2) &= \max(1 - \frac{\beta}{\rho \|\mathbf{K}_2 \mathbf{u}_2 + \boldsymbol{\lambda}_2\|}, \mathbf{0}) \circ (\mathbf{K}_2 \mathbf{u}_2 + \boldsymbol{\lambda}_2), \\ \text{prox}_{\frac{f_3}{\rho}}(\mathbf{K}_3 \mathbf{u}_3 + \boldsymbol{\lambda}_3) &= \begin{cases} \mathbf{K}_3 \mathbf{u}_3 + \boldsymbol{\lambda}_3 & : \|\mathbf{K}_3 \mathbf{u}_3 + \boldsymbol{\lambda}_3\|_2^2 \leq 1, \\ \frac{\mathbf{K}_3 \mathbf{u}_3 + \boldsymbol{\lambda}_3}{\|\mathbf{K}_3 \mathbf{u}_3 + \boldsymbol{\lambda}_3\|_2} & : \text{else.} \end{cases} \end{aligned} \quad (\text{A-8})$$

REFERENCES

- Al-Madani, M., U. b. Waheed, and M. Masood, 2019, Fast and accurate dictionary learning for seismic data denoising using convolutional sparse coding, *in* SEG Technical Program Expanded Abstracts 2019: Society of Exploration Geophysicists, 4645–4649.
- Beckouche, S., and J. Ma, 2014, Simultaneous dictionary learning and denoising for seismic data: *Geophysics*, **79**, no. 3, A27–A31.
- Bekara, M., and M. Van der Baan, 2009, Random and coherent noise attenuation by empirical mode decomposition: *Geophysics*, **74**, no. 5, V89–V98.
- Bonar, D., and M. Sacchi, 2012, Denoising seismic data using the nonlocal means algorithm: *Geophysics*, **77**, no. 1, A5–A8.
- Boyd, S., N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., 2011, Distributed optimization and statistical learning via the alternating direction method of multipliers: *Foundations and Trends® in Machine learning*, **3**, no. 1, 1–122.
- Bristow, H., A. Eriksson, and S. Lucey, 2013, Fast convolutional sparse coding: *Computer Vision and Pattern Recognition (CVPR)*, 2013 IEEE Conference on, IEEE, 391–398.
- Chanerley, A., and N. Alexander, 2002, An approach to seismic correction which includes wavelet de-noising: *Proceedings of the sixth conference on Computational structures technology*, Civil-Comp Press, 107–108.
- Chen, Y., 2016, Dip-separated structural filtering using seislet transform and adaptive empirical mode decomposition based dip filter: *Geophysical Journal International*, **206**, no. 1, 457–469.
- Chen, Y., J. Ma, and S. Fomel, 2016, Double-sparsity dictionary for seismic noise attenuation: *Geophysics*, **81**, no. 2, V103–V116.
- Choudhury, B., R. Swanson, F. Heide, G. Wetzstein, and W. Heidrich, 2017, Consensus con-

- volutional sparse coding: Proceedings of the IEEE International Conference on Computer Vision, 4280–4288.
- Elad, M., and M. Aharon, 2006, Image denoising via sparse and redundant representations over learned dictionaries: IEEE Transactions on Image processing, **15**, no. 12, 3736–3745.
- Gu, S., W. Zuo, Q. Xie, D. Meng, X. Feng, and L. Zhang, 2015, Convolutional sparse coding for image super-resolution: Proceedings of the IEEE International Conference on Computer Vision, 1823–1831.
- Guitton, A., 2002, Coherent noise attenuation using inverse problems and prediction-error filters: First Break, **20**, no. 3, 161–167.
- Heide, F., W. Heidrich, and G. Wetzstein, 2015, Fast and flexible convolutional sparse coding: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5135–5143.
- Hennenfent, G., and F. J. Herrmann, 2006, Seismic denoising with nonuniformly sampled curvelets: Computing in Science Engineering, **8**, no. 3, 16–25.
- Ibrahim, A., and M. Sacchi, 2014, Eliminating blending noise using fast apex shifted hyperbolic radon transform: Presented at the 76th EAGE Conference and Exhibition 2014.
- Jas, M., T. D. La Tour, U. Simsekli, and A. Gramfort, 2017, Learning the morphology of brain signals using alpha-stable convolutional sparse coding: Advances in Neural Information Processing Systems, 1099–1108.
- Kaplan, S. T., M. D. Sacchi, and T. J. Ulrych, 2009, Sparse coding for data-driven coherent and incoherent noise attenuation: 2009 SEG Annual Meeting, Society of Exploration Geophysicists, 3327–3331.
- Li, C., Y. Zhang, and C. C. Mosher, 2018, An unsupervised learning method for residual seismic signal recovery, *in* SEG Technical Program Expanded Abstracts 2018: Society of

- Exploration Geophysicists, 1996–2000.
- Li, J., S. Hanafy, Z. Liu, and G. T. Schuster, 2019, Wave equation dispersion inversion of love waves: *Geophysics*, **84**, no. 5, R693–R705.
- Li, S., X. Yang, H. Liu, Y. Cai, and Z. Peng, 2020, Seismic data denoising based on sparse and low-rank regularization: *Energies*, **13**, no. 2, 372.
- Liu, G.-C., X.-H. Chen, J.-Y. Li, J. Du, and J.-W. Song, 2011, Seismic noise attenuation using nonstationary polynomial fitting: *Applied Geophysics*, **8**, no. 1, 18–26.
- Liu, Z., Y. Chen, and S. Gerard, 2020, Deep convolutional neural network and sparse least squares migration: *Geophysics*, **85**, no. 4, WA241–WA253.
- Liu, Z., J. Li, S. M. Hanafy, and G. T. Schuster, 2019, 3D wave-equation dispersion inversion of Rayleigh waves: *Geophysics*, **84**, no. 5, R673–R691.
- Liu, Z., and G. Schuster, 2019, Multilayer sparse LSM= deep neural network, *in* SEG Technical Program Expanded Abstracts 2019: Society of Exploration Geophysicists, 2323–2327.
- Papayan, V., Y. Romano, and M. Elad, 2017a, Convolutional neural networks analyzed via convolutional sparse coding: *The Journal of Machine Learning Research*, **18**, no. 1, 2887–2938.
- Papayan, V., Y. Romano, J. Sulam, and M. Elad, 2017b, Convolutional dictionary learning via local processing: *Proceedings of the IEEE International Conference on Computer Vision*, 5296–5304.
- Parikh, N., and S. Boyd, 2014, Proximal algorithms: *Foundations and Trends in optimization*, **1**, no. 3, 127–239.
- Peter, S., E. Kirschbaum, M. Both, L. Campbell, B. Harvey, C. Heins, D. Durstewitz, F. Diego, and F. A. Hamprecht, 2017, Sparse convolutional coding for neuronal assembly

- detection: *Advances in Neural Information Processing Systems*, 3675–3685.
- Serrano, A., F. Heide, D. Gutierrez, G. Wetzstein, and B. Masia, 2016, Convolutional sparse coding for high dynamic range imaging: *Computer Graphics Forum*, Wiley Online Library, 153–163.
- Wohlberg, B., 2016, Boundary handling for convolutional sparse representations: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 1833–1837.
- Xiong, J., P. Richtárik, and W. Heidrich, 2019, Stochastic convolutional sparse coding: arXiv preprint arXiv:1909.00145.
- Yilmaz, Ö., 2001, *Seismic data analysis: Processing, inversion, and interpretation of seismic data*: Society of Exploration Geophysicists.
- Yu, S., J. Ma, and W. Wang, 2019, Deep learning for denoising: *Geophysics*, **84**, no. 6, V333–V350.
- 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.
- Zisselman, E., J. Sulam, and M. Elad, 2019, A local block coordinate descent algorithm for the CSC model: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8208–8217.

LIST OF FIGURES

1	Workflow of CSC for noise attenuation.	23
2	Marmousi P-wave velocity model used for computing the synthetic data. . .	24
3	(a) Synthetic data; (b) noisy data; the denoised results by the (c) CSC and (d) traditional SC methods based on the K-SVD algorithm. The red rectangle in (a) is selected for comparison in Figure 4.	25
4	Comparison of the learned filters from the noisy data by (a) CSC and (b) traditional SC based on the K-SVD method. Here, the red rectangles show the filters with similar features.	26
5	(a) Synthetic data from Marmousi model and (b) its noisy data with SNR of -1.76; (c) the denoised data with SNR of 14.5 and (d) the residual between (b) and (c).	27
6	Fifty learned filters by CSC with size of 55×55 from five high-SNR CSGs.	28
7	Denoised data by setting $\beta =$ (a) 0.2 and (b) 5 and their corresponding residuals in (c) and (d), respectively.	29
8	(a) One CSG of the training data sets which are from the near offset of the Qademah fault seismic data with offset range from 0.2 km to 1.3 km, where the ground roll has been muted; (b) 30 learned filters with the size of 21×21 by CSC.	30
9	(a) The noisy data from the far offset of the Qademah fault seismic data with offset range from 0.8 km to 1.9 km; the denoised data by (a) all filters, and selected filters excluding those indicated by (b) red and (c) red and yellow boxes in Figure 8b.	31
10	Residuals of the denoised data shown in Figure 9b-9d.	32
11	(a) Near-offset data with surface waves, (b) denoised data, and (c) its residual.	33
12	Projection of $\mathbf{b}$ on the convex set $\ \mathbf{y}\ _2^2 \leq 1$	34

Figure 1: Workflow of CSC for noise attenuation.

Figure 2: Marmousi P-wave velocity model used for computing the synthetic data.

Figure 3: (a) Synthetic data; (b) noisy data; the denoised results by the (c) CSC and (d) traditional SC methods based on the K-SVD algorithm. The red rectangle in (a) is selected for comparison in Figure 4.

(a) **CSC Filters**

(b) **KSVD Filters**

Figure 4: Comparison of the learned filters from the noisy data by (a) CSC and (b) traditional SC based on the K-SVD method. Here, the red rectangles show the filters with similar features.

Figure 5: (a) Synthetic data from Marmousi model and (b) its noisy data with SNR of -1.76; (c) the denoised data with SNR of 14.5 and (d) the residual between (b) and (c).

Learned Features

Figure 6: Fifty learned filters by CSC with size of 55×55 from five high-SNR CSGs.

Figure 7: Denoised data by setting $\beta =$ (a) 0.2 and (b) 5 and their corresponding residuals in (c) and (d), respectively.

Figure 8: (a) One CSG of the training data sets which are from the near offset of the Qademah fault seismic data with offset range from 0.2 km to 1.3 km, where the ground roll has been muted; (b) 30 learned filters with the size of 21×21 by CSC.

Figure 9: (a) The noisy data from the far offset of the Qademah fault seismic data with offset range from 0.8 km to 1.9 km; the denoised data by (a) all filters, and selected filters excluding those indicated by (b) red and (c) red and yellow boxes in Figure 8b.

Figure 10: Residuals of the denoised data shown in Figure 9b-9d.

Figure 11: (a) Near-offset data with surface waves, (b) denoised data, and (c) its residual.

Figure 12: Projection of $\mathbf{b}$ on the convex set $\|\mathbf{y}\|_2^2 \leq 1$.