Quality control of deep generator priors for statistical seismic inverse problems

Zhilong Fang[1,3], Hongjian Fang[2,3], Laurent Demanet[1,2,3]
[1]*Department of Mathematics, Massachusetts Institute of Technology*
[2]*Department of Earth, Atmospheric, and Planetary Sciences, Massachusetts Institute of Technology*
[3]*Earth Resource Laboratory, Massachusetts Institute of Technology*

SUMMARY

Prior information is essential for statistical seismic inverse problems to quantify uncertainties of unknown subsurface parameters. Conventional prior knowledge is based on empirical observations of subsurface structures like the smoothness of the subsurface image. However, such hand-designed prior knowledge is too generic to describe detailed subsurface structures. Deep generator priors are a recent encouraging development, where a high-quality prior can be crafted solely from existing data and a deep neural network. Such deep generators can create detailed subsurface models from Gaussian vectors in a low-dimensional latent space. Alongside this potential advantage, comes the major risk that the prior might be too restrictive to give physically meaningful solutions a high enough likelihood. In this work, we attempt to mitigate this risk by presenting a measure to evaluate the quality of the learned deep generator prior, which can then be leveraged to tune its hyperparameters. Given a testing model and a generator, we first find the best generated model that minimizes the ℓ_2-norm misfit between the generated and testing models. To measure the quality of the generator, we suggest two criteria: (1) the ℓ_2-norm model error should be small; and (2) the ℓ_2-norm of the corresponding latent vector should be bounded and within a pre-defined reference range. Numerical examples show that a good prior generator in the sense of the proposed measure can help produce more accurate results for statistical seismic inverse problems.

INTRODUCTION

Geophysicists require statistical information of subsurface parameters to understand the interior of the Earth. For this purpose, statistical approaches, and in particular the Bayesian inference method (Kaipio and Somersalo, 2006), are desirable and necessary. The Bayesian inference method has been applied to many geophysical problems (Tarantola and Valette, 1982b; Fang et al., 2018). It considers unknown parameters as random variables and aims at a posterior probability density function (PDF) that incorporates all available statistical information from both the observations in a likelihood PDF and researchers' prior knowledge in a prior PDF. These statistics reflect the degree of confidence about the unknown parameters and allow researchers to identify areas with high/low reliability in the model.

The prior knowledge plays a key role in the Bayesian inverse problem. Conventional prior knowledge is hand-crafted based on researchers' empirical observations about subsurface structures. For instances, because subsurface structure images are generally piecewise smooth and sparse after wavelet or curvelet transformations, image priors that constrain the sparsity of wavelet or curvelet coefficients (Ying et al., 2005; Tu and Herrmann, 2015) or spatial gradients (Haber et al., 2000) are widely utilized. While these hand-crafted priors can regularize the unknown parameters when solving deterministic inverse problems, they are usually too generic, in that prior models generated with these prior PDF cannot describe detailed structures.

Recent developments in deep convolutional neural networks (DCNN) (Krizhevsky et al., 2012) provide researchers with a new way to design the prior knowledge, i.e. a learning-based prior generator. Different from hand-crafted priors, learning-based approaches train DCNNs that learn features from existing samples. The trained DCNN is designed to generate images sharing similar distributions of natural samples. Mosser et al. (2020) and Herrmann et al. (2019) show the potential applications of applying DCNN to seismic inverse problems like full waveform inversion and least-squared reverse time migration.

Despite the potential advantages of learning-based prior generators in producing detailed subsurface models, a quantitative measure that evaluates the quality of different generators remain unresolved. This issue is consequential: the new point of view represents a swing from human-made priors that are generally too loose, to network-based priors that might be way too restrictive. In this work, we propose a quality measure for deep generators that is at the same time computable and informative of the quality of the resulting prior.

Following Mosser et al. (2020), we use a deep generative adversarial network (DGAN) (Goodfellow et al., 2014) as our generator and train it with given velocity models. One appealing property of DGAN for the Bayesian inversion lies in its potential capability in generating perceptually appealing high-dimensional images from a Gaussian vector in a low-dimensional latent parameter space (Ledig et al., 2017). In addition, the prior distribution in the latent space is a standard normal distribution. We evaluate the trained generator with several models from a testing velocity set. Our quality measure is based on two straightforward requirements for a good generator: (1) the testing model should be in the range of the generator, or the difference between the best generated and testing models should be small; (2) since the prior distribution is a standard normal distribution, the corresponding latent vector of the matching generated model should have a moderate ℓ_2-norm as a measure of plausibility. We use these two requirements to derive our quality measure.

METHODOLOGY

Bayesian inference

The Bayesian inference method aims at a comprehensive statistical description of the unknown parameters. It constructs a posterior PDF of the model parameters $\mathbf{m}$, integrating statistical information from the forward map $F(\mathbf{m})$, observed data $\mathbf{d}$, and researcher's prior knowledge. According to the Bayes' law,

the posterior PDF $\rho_{\text{post}}(\mathbf{m}|\mathbf{d})$ of $\mathbf{m}$ given $\mathbf{d}$ is proportional to the product of a likelihood PDF $\rho_{\text{like}}(\mathbf{d}|\mathbf{m})$ of $\mathbf{d}$ given $\mathbf{m}$ and a prior PDF $\rho_{\text{prior}}(\mathbf{m})$ of $\mathbf{m}$ as follows:

$$\rho_{\text{post}}(\mathbf{m}|\mathbf{d}) \propto \rho_{\text{like}}(\mathbf{d}|\mathbf{m})\rho_{\text{prior}}(\mathbf{m}). \quad (1)$$

The prior PDF $\rho_{\text{prior}}(\mathbf{m})$ describes one's prior knowledge and beliefs in the unknown model parameters, and the likelihood PDF $\rho_{\text{like}}(\mathbf{d}|\mathbf{m})$ describes the probability of observing data $\mathbf{d}$ given the model $\mathbf{m}$. With the posterior PDF in hands, we can extract statistics of interests including the maximum a posterior (MAP) estimate, model covariance matrix, model standard deviation (STDev), and confidence interval of $\mathbf{m}$.

Deep GAN model prior generator

The selection of the prior PDF plays an important role in the statistical inversion. Different from the conventional hand-crafted priors, Mosser et al. (2020) shows that we can use a DGAN to train a prior generator directly from existing subsurface images. Here, we use a least-squares DGAN (LS-DGAN) (Mao et al., 2017) to design such a generator. Given a training set $\mathscr{M}$, LS-DGAN aims to generate artificial samples sharing the same statistics as the training samples. Similar to standard DGANs, LS-DGAN consists of two neural networks namely the generator $G(\mathbf{x};\Theta_G)$ and the discriminator $D(\mathbf{m};\Theta_D)$, which are parameterized by vectors Θ_G and Θ_D. The generator creates candidates of interests from vectors $\mathbf{x}$'s obeying a distribution $\mathscr{X}$ in a low-dimensional latent space, while the discriminator uses an *a*-*b* coding scheme to distinguish these generated candidates from the true data distribution. Parameters a and b denote the labels for generated samples and real samples. The discriminator tries to decrease the error rate of the discriminative network, while the generator tries to increase the error rate. To train the two networks, LS-DGAN solves the following *minmax* optimization problem:

$$\begin{aligned}\min_{\Theta_G}\max_{\Theta_D}&\mathbb{E}_{\mathbf{m}\sim\mathscr{M}}[D(\mathbf{m};\Theta_D)-a]^2\\ &+\mathbb{E}_{\mathbf{x}\sim\mathscr{X}}[D(G(\mathbf{x};\Theta_G);\Theta_D)-b]^2.\end{aligned} \quad (2)$$

where the expectation for $\mathbf{m}$ is over all training samples in $\mathscr{M}$, and the expectation for $\mathbf{x}$ is over the distribution $\mathscr{X}$. In general, $\mathscr{X}$ can be a set of images or any reasonable predefined distributions. Here, we select $\mathscr{X}$ to be a standard Gaussian distribution $\mathscr{N}(0,\mathbf{I})$ for a series of computational advantages in the following statistical inversion.

The generator $G(\mathbf{x};\Theta_G)$ defines an n_{l}-manifold in $\mathbb{R}^{n_{\text{g}}}$, where $n_{\text{l}} \ll n_{\text{g}}$. With the additional assumption that the observed data contain additive Gaussian noise $\varepsilon \sim \mathscr{N}(0,\Sigma)$ with a covariance matrix Σ, we can reformulate the original Bayesian inversion problem on the n_{l}-manifold as follows,

$$\rho_{\text{post}}(\mathbf{x}|\mathbf{d}) \propto \exp(-\frac{1}{2}\|F(G(\mathbf{x};\Theta_G))-\mathbf{d}\|^2_{\Sigma^{-1}} - \frac{1}{2}\|\mathbf{x}\|^2). \quad (3)$$

Compared to the original posterior PDF in Equation 1, studying the new posterior PDF $\rho_{\text{post}}(\mathbf{x}|\mathbf{d})$ has three advantages. First, we reduce the dimensionality from n_{g} to n_{l}, which makes the new Bayesian inversion problem less suffer from the curse of dimensionality. Secondly, the Gaussian prior distribution is directly from the definition of the LS-DGAN generator without any hand designs. Finally, the Gaussian prior distribution enables us to apply the computationally efficient Markov chain Monte Carlo (McMC) method – preconditioner Cranks Nickos method (Cotter et al., 2013) to sample the posterior PDF.

Quantitative measure for quality control

To derive the quantitative measure for the quality control of deep generators, we notice two facts for a good generator: (1) the testing model should be in the range of the generator or at least close to the range of the generator; (2) the latent vector of the best generated model matching the testing model should lie in the high probability zone of the prior distribution. To evaluate both requirements, given a testing model $\mathbf{m}_{\text{test}}$, we first find its best representative by solving the following problem:

$$\mathbf{x}_{\text{test}} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{m}_{\text{test}} - G(\mathbf{x};\Theta_G)\|^2. \quad (4)$$

The first requirement asks for the relative error $\text{e} = \|\mathbf{m}_{\text{test}} - G(\mathbf{x}_{\text{test}};\Theta_G)\|/\|\mathbf{m}_{\text{test}}\|$ to be small. The second requirement needs to utilize the property of the prior distribution on $\mathbf{x}$. Since the prior distribution is a standard normal distribution, a vector lies in a high probability zone should have a small ℓ_2-norm magnitude. This asks for $\|\mathbf{x}_{\text{test}}\|$ smaller than a reference value ν. Moreover, the latent vector with bounded ℓ_2-norm magnitude also ensures that the generator is not overfitting as a result of ill-conditioning. A good choice for ν would be the average ℓ_2-norm of all possible samples from the prior distribution. Finally, there is the issue that $G(\mathbf{x};\Theta_G)$ is nonconvex as a function of $\mathbf{x}$, which is mitigated – empirically and to a good extent – by considering a large number of random initializers.

SEISMIC INVERSION APPLICATIONS

We assess the effectiveness of the proposed quality measure by conducting Bayesian seismic inversions with traveltime tomography (TT) (Aki et al., 1977) and full waveform inversion (FWI) (Tarantola and Valette, 1982a). To conduct experiments, we first train the LS-DGAN using velocity slices extracted from the 3D Overthrust model. Then, we conduct statistical inversions with the trained generators on a different velocity slice extracted from the Overthrust model.

Training the network architecture

Network architectures. We design three generators with different numbers of layers and apply the proposed measure to evaluate their qualities. The three generative networks (GN) consist of eight, six, and four block layers, respectively. To simplify the notation, we name them GN-8, GN-6, and GN-4, respectively. We also design a discriminative network with three block layers. Following Mao et al. (2017), each block includes parts of the following operations or networks: fully-connected networks, deconvolutional neural networks, convolutional neural networks, batch normalizations, dropout operations, leaky ReLU activations, and tanh activation functions. The generative networks map a random vector $\mathbf{x} \sim \mathscr{N}(0,\mathbf{I}_{50\times 50})$ to a 64×64 velocity image, while the discriminative network maps a 64×64 velocity image to a real number that aims to fit the labels $a=0$ and $b=1$ for the real and generated samples.

Training. We train the generative and discriminative networks with velocity models extracted from the 3D Overthrust model. We extract 19,762 velocity models with the size of 64×64 from the 3D volume and rotate them with random angles, yielding 79,048 velocity models. We use these models to train the generators and discriminator. We use the Adam optimizer to

train the networks, and select the momentum parameter $\gamma = 0.9$, mini-batch size $L_b = 256$, and initial rate $\mu = 1e\text{-}4$.

Sample comparison of different generators

We compare the performance of the three generators in producing samples. Figure 1 shows a comparison of the images from the training set and the three generative networks. Visually, it is difficult to evaluate these images, since they all present some layered structures as shown in the training set. To further evaluate the three generators, we use each generator to produce 50,000 samples and compute the point-wise STDev and probability density of the 50,000 samples. Figure 2 shows the comparison of the STDevs of the training samples and generated samples of the three generators. The STDevs corresponding to GN-8 and GN-6 match with that of the training samples much better than the one of GN-4. Figure 3 shows the comparison of the point-wise probability densities at three different locations. Clearly, the probability densities of GN-8 and GN-6 coincide with that of the training samples much better than the one of GN-4. Both comparisons imply that increasing the number of layers can yield a generator that can capture the distribution of the training set more accurately.

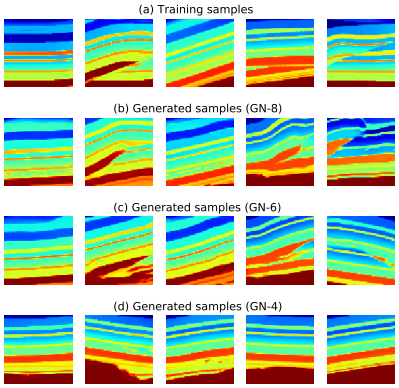


Figure 1: Comparison of images from (a) training set, (b) GN-8, (c) GN-6, and (d) GN-4.

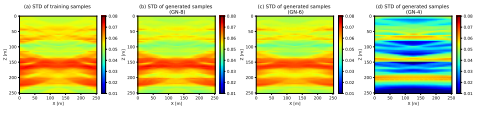


Figure 2: Comparison of STDevs of (a) training samples, and samples generated by (b) GN-8, (c) GN-6, and (d) GN-4.

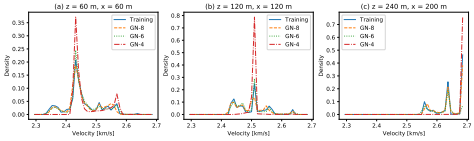


Figure 3: Comparison of probability densities of training samples (solid) and samples generated by GN-8 (dashed), GN-6 (dotted), and GN-4 (dashdot) at locations: (a) $z = 60$m,$x = 60$m; (b) $z = 120$m,$x = 120$m; and (c) $z = 240$m,$x = 200$m.

Generator quantitative evaluation

We use the proposed quality measure to evaluate the three generators. We first select three testing models $\mathbf{m}_{\mathrm{t},i}\,(1 \le i \le 3)$ from the Overthrust model, which are not in the training set. Then for the jth generator ($1 \le j \le 3$) and each $\mathbf{m}_{\mathrm{t},i}$, we find the best representative $\mathbf{x}_{j,i}$ in the latent space by solving the optimization problem in Equation 4. To solve this optimization problem, we first generate 1000 samples and select the one $\mathbf{x}_{\mathrm{ini},j,i}$ that produces the least ℓ_2-norm misfit. We start with $\mathbf{x}_{\mathrm{ini},j,i}$ and use 100,000 gradient descent steps to find the optimal solution. Figure 4 shows the three selected models and their corresponding best generated models $\{\mathbf{m}_{\mathrm{g},j,i}\}_{1\le i,j\le 3}$ produced by the three generators. Clearly, the generated models given by GN-8 are much closer to the testing models, compared to GN-6 and GN-4.

Figure 5 plots the relative model error and logarithm ℓ_2-norm of the latent vector for each generated model. For a good prior generator, we expect that all the points should be at the bottom of the Gaussian zone (gray area). Clearly, GN-8 produces results with minimal relative model errors. Meanwhile, the $\|\mathbf{x}_{j,i}\|$ of the GN-8 results are closer to the Gaussian zone. This test illustrates that compared to GN-6 and GN-4, GN-8 has a stronger capability in representing testing models, which implies a stronger generality of GN-8.

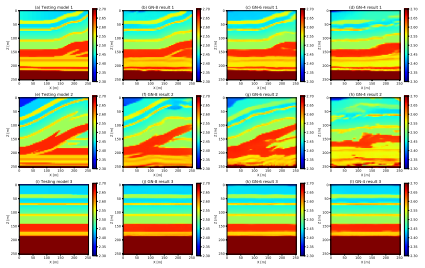


Figure 4: (a),(e), and (i) The three selected testing models; (b), (f) and (j) the three generated models of GN-8; (c), (g) and (k) the three generated models of GN-6; (d), (h) and (l) the three generated models of GN-4;

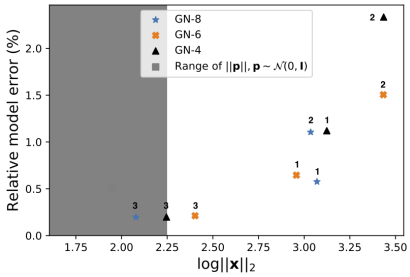


Figure 5: ℓ_2-norm relative model error $\frac{\|\mathbf{m}_{\mathrm{t},i}-\mathbf{m}_{\mathrm{g},j,i}\|_2}{\|\mathbf{m}_{\mathrm{t},i}\|_2}$ between the testing models $\{\mathbf{m}_{\mathrm{t},i}\}$ and the generated models $\{\mathbf{m}_{\mathrm{g},j,i}\}$ of GN-8 ($\star$), GN-6 ($\times$) and GN-4 (Δ) v.s. logarithm ℓ_2-norm of their latent vectors $\mathbf{x}_{j,i}$. The gray area represents the range of the ℓ_2-norm of standard Gaussian random vectors with the size of the latent vector. The indices 1, 2, and 3 label different testing model scenarios.

Statistical inversion

We conduct the Bayesian inversion with a testing velocity model $\mathbf{m}_{\mathrm{t}}$ extracted from the 3D Overthrust model (Figure 6a) that is

not in the training set. Since both the testing and training models are extracted from the 3D Overthrust model, their spatial distributions should share a strong similarity and the prior generator should provide good prior information for the inversion.

TT example. We first conduct the statistical inversion with TT. We mimic a cross-hole case, where 30 sources ($\star$) are located at the left side of the model and 60 receivers (Δ) are located at the right side of the model as shown in Figure 6a. The model size is 258 m $\times$ 258 m. We rescale the velocity model so that the range of the velocity is $[2.4\text{km/s}, 2.7\text{km/s}]$. The observed data contain 0.5% additive Gaussian noise.

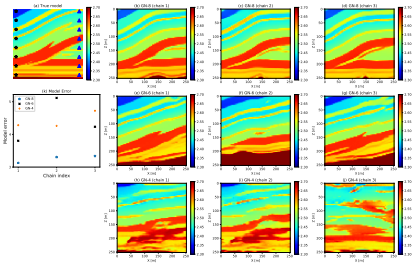


Figure 6: (a) True model with indicated source ($\star$) and receiver (Δ) locations; MAP estimates of the three chains associated with (b) - (d) GN-8, (e) - (g) GN-6, and (h) - (j) GN-4.

We conduct Bayesian inversions with the three generators to compare their performances. For each generator, we use three different starting points to draw three Markov chains with 100,000 samples to study the influence of starting points to inversion results. We use the pCN method to produce the 9 Markov chains with a burn-in period of 30,000 steps. We select the step size for pCN to ensure an average acceptance ratio of 0.4.

Figures 6b - 6j show the MAP estimates of the 9 Markov chains, and Figure 6k shows the ℓ_2-norm model error between each MAP estimate and the true model. From Figures 6, we can observe that the GN-8 outperforms GN-6 and GN-4 from the perspective of matching structures. The MAP estimates of GN-8 matches most layered structures in the true model, despite some mismatches at the boundary of the model. Figure 6k shows that GN-8 produces MAP estimates with the minimal model errors. Moreover, the variation of the three MAP estimates of GN-8 is visually smaller than that of GN-6 and GN-4.

We compute the point-wise STDevs for all the 9 chains. For each generator, we also compute the STDev from the ensemble of the three individual chains. Figure 7 shows all the computed STDevs. Overall, we can observe large STDevs around the boundary of layers and small STDevs inside layers. Comparing STDevs of individual chains with STDevs of their corresponding ensembles, we can observe that the difference between STDevs of individual chains and the ensemble chain associated with GN-8 is significantly smaller than those associated with GN-6 and GN-4. This result coincides with the observation that MAP estimates of GN-8 have the smallest variation.

FWI example. We use GN-8 and conduct the second statistical inversion with the application of FWI. We use the same velocity model as the one used in the first example. We also mimic a cross-hole case, in which we place 8 sources at the left side of the model and place 64 receivers at the right side of the model. Figure 8a shows the true model and locations of sources and receivers. We use a Ricker wavelet centered at 10Hz to simulate the data with 1% additive Gaussian noise. We use the same strategy as mentioned in the first example to select the starting point and use the pCN method to generate $100,000$ samples with a burn-in period of 30,000 steps. We compute statistics of interests with the remaining 70,000 samples. Figure 8b shows the MAP estimate of the 70,000 samples. Clearly, the MAP estimate matches the true model quite well. As a comparison, Figure 8c shows the result of the conventional FWI using the same starting point. We can observe that the MAP estimate obtained by the proposed method presents a better resolution compared to the conventional FWI that directly works on the physical domain. Figure 8d illustrates the point-wise STDev of the 70,000 samples. Similar to the result of traveltime tomography, we can observe high STDevs at boundaries of layers and low STDevs inside layers.

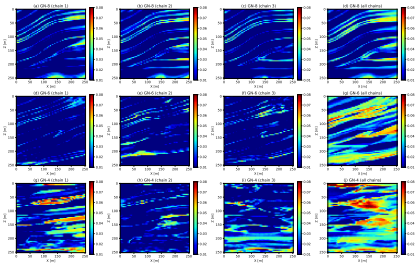
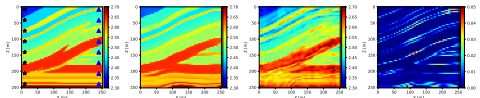


Figure 7: STDevs of the three chains and their ensembles associated with (a) - (d) GN-8 , (e) - (g) GN-6, and (g) - (j) GN-4.



(a) True (b) MAP (c) CFWI (d) STDev

Figure 8: (a) True model for FWI. (b) Result of the conventional FWI (CFWI). (c) MAP of the statistical inversion. (d) STDev.

CONCLUSIONS

We present a quantitative measure to evaluate the quality of the recently developed deep generator prior for statistical seismic inverse problems. The proposed quality control measure considers both model errors in the original image domain and the ℓ_2-norm of the corresponding latent vector. Numerical examples illustrate that a good prior generator in the sense of the proposed measure is able to produce better inversion results when contrasted with a prior that is not carefully chosen. Although we use testing models to evaluate the generators, for situations that testing models are not available, we can leverage unused validation models as an alternative to evaluate the learned generators.

ACKNOWLEDGE

The authors acknowledge the funding and support provided by ExxonMobil Research. Dr. Laurent Demanet is also supported by AFOSR grant FA9550-17-1-0316.

REFERENCES

Aki, K., A. Christoffersson, and E. S. Husebye, 1977, Determination of the three-dimensional seismic structure of the lithosphere: Journal of Geophysical Research, **82**, 277–296.

Cotter, S. L., G. O. Roberts, A. M. Stuart, D. White, et al., 2013, Mcmc methods for functions: modifying old algorithms to make them faster: Statistical Science, **28**, 424–446.

Fang, Z., C. Da Silva, R. Kuske, and F. J. Herrmann, 2018, Uncertainty quantification for inverse problems with weak partial-differential-equation constraints: Geophysics, **83**, R629–R647.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, 2014, Generative adversarial nets: Advances in neural information processing systems, 2672–2680.

Haber, E., U. M. Ascher, and D. Oldenburg, 2000, On optimization techniques for solving nonlinear inverse problems: Inverse problems, **16**, 1263.

Herrmann, F. J., A. Siahkoohi, and G. Rizzuti, 2019, Learned imaging with constraints and uncertainty quantification: arXiv preprint arXiv:1909.06473.

Kaipio, J., and E. Somersalo, 2006, Statistical and computational inverse problems: Springer Science & Business Media.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012, Imagenet classification with deep convolutional neural networks: Advances in neural information processing systems, 1097–1105.

Ledig, C., L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., 2017, Photo-realistic single image super-resolution using a generative adversarial network: Proceedings of the IEEE conference on computer vision and pattern recognition, 4681–4690.

Mao, X., Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, 2017, Least squares generative adversarial networks: Proceedings of the IEEE International Conference on Computer Vision, 2794–2802.

Mosser, L., O. Dubrule, and M. J. Blunt, 2020, Stochastic seismic waveform inversion using generative adversarial networks as a geological prior: Mathematical Geosciences, **52**, 53–79.

Tarantola, A., and B. Valette, 1982a, Generalized nonlinear inverse problems solved using the least squares criterion: Reviews of Geophysics, **20**, 219–232.

———, 1982b, Inverse problems = quest for information: Journal of Geophysics, **50**, 159–170.

Tu, N., and F. J. Herrmann, 2015, Fast imaging with surface-related multiples by sparse inversion: Geophysical Journal International, **201**, 304–317.

Ying, L., L. Demanet, and E. Candes, 2005, 3D discrete curvelet transform: Wavelets XI, International Society for Optics and Photonics, 591413.