SEISMIX 2020

Seismic Inversion by Newtonian Machine Learning

Yuqing Chen[1], Erdinc Saygin[1], Gerard Schuster[2]
[1]Deep Earth Imaging Future Science Platform, CSIRO, [2]King Abdullah University of Science and Technology

Summary

We present a wave-equation inversion method that inverts skeletonized data for the subsurface velocity model. The skeletonized representation of these seismic traces consist of the low rank latent-space variables predicted by a well-trained autoencoder neural network. The input to the autoencoder consist of the recorded common shot gathers, and the implicit function theorem is used to determine the perturbation of the skeletonized data with respect to the velocity perturbation. The final velocity model is the one that best predicts the observed latent-space parameters. We denote this hybrid inversion method as Newtonian machine learning (NML) inversion because it inverts for the model parameters by combining the deterministic laws of Newtonian physics with the statistical capabilities of machine learning.

Introduction

Full waveform inversion (FWI) has been shown to accurately invert seismic data for high-resolution velocity model. However, the success of FWI heavily relies on a good initial model that is close to the true model, otherwise, cycle-skipping problem will trap FWI in a local minimum. The main reason non-linear inversion gets stuck in a local minimum is that the data are very complex (i.e, wiggly in time), which means that the objective function is very complex and characterized by many multiple minimums. To avoid this problem, on way is to skeletonize the complex data into a much simpler form and use the skeletonized information as a new misfit function for inversion.

The autoencoder neural network is an unsupervised deep learning method that is trained for dimensionality reduction. An autoencoder maps the data into a lower dimensional space by extracting the data's most important features, these features are also denoted as the skeletonized representation of the input data. In this abstract, we feed the observed traces into the autoencoder to get their corresponding low-dimension representation. We build the misfit function as the sum of the squared differences between the observed and predicted encoded value

$$\epsilon = \sum_s \sum_g \left\| z_{obs}(g|s) - z_{syn}(g|s) \right\|_2^2,$$

Where $z_{obs}(g|s)$ and $z_{syn}(g|s)$ represent the encoded value of the observed and synthetic traces by autoencoder. To compute the gradient with respect to the model parameters such as the velocity in each pixel, we use this implicit function theorem to compute the perturbation of the skeletonized information with respect to the velocity. The high-level strategy for inverting the skeletonized latent variable is summarized in Figure 1, where L corresponds to the forward modeling operator of the governing equations, such as wave-equation

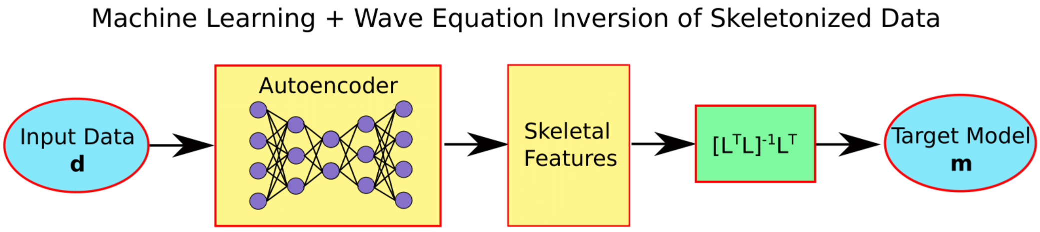

Figure1: The high-level strategy for Newtonian machine learning inversion.

Numerical Example

We test this NML inversion method on a layered model and a crosswell acquisition geometry. Figures 2a and 2b show the true and initial velocity model, respectively. A fixed-spread crosswell acquisition geometry is deployed where the source and receiver wells are located at x = 10 and x = 1000 m, respectively.

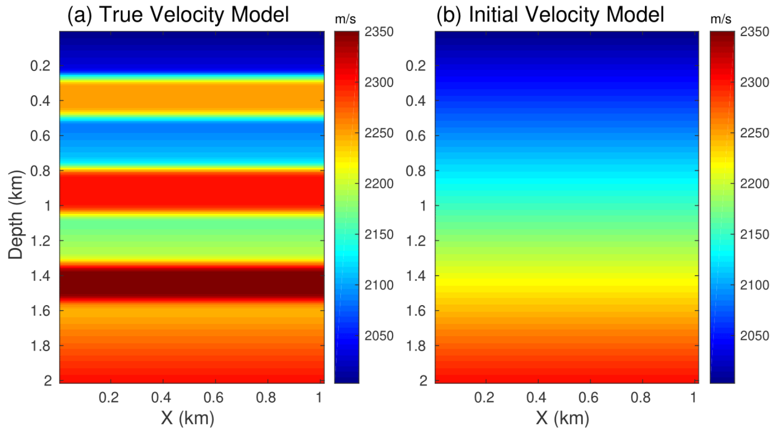

Figure2: The (a) true and (b) initial velocity model.

The training set includes 4000 observed seismic traces and trained by an three layer autoencoder network with Tanh activation function. After the autoencoder neural network is well trained, we input the synthetic traces generated in each iteration to get their encoded values. Therefore the skeletonized misfit and gradient can be calculated in order to update the velocity model. Figure 3a shows the inverted result which successfully recovers the three high-velocity layers and two of it's vertical profiles are compared in Figures 3b and 3c, respectively, where the blue, red and black lines represent the velocity profiles from the initial, true an inverted velocity models.

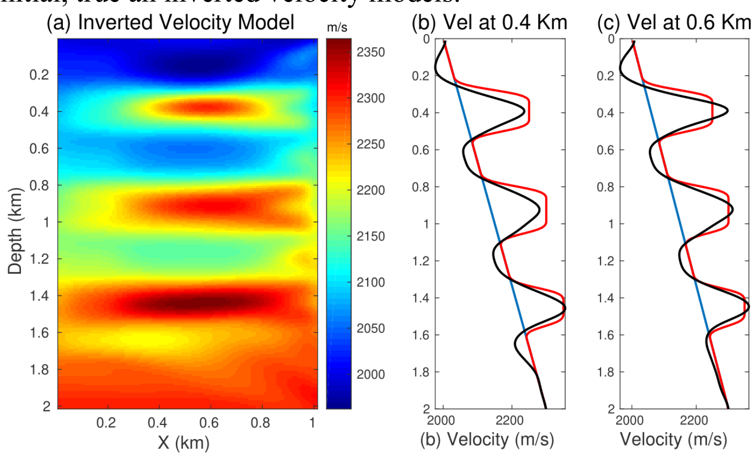

Figure3: The (a) inverted model and the velocity comparison at (b) x = 0.4 km and (c) x = 0.6 km.