Neural Nets as a Filter in Two-Stage Markov Chain Monte Carlo for Velocity Estimation and Uncertainty Quantification

SIAM Conference on Mathematical & Computational Issues in the Geosciences (GS19)

Georgia K. Stuart*, Susan E. Minkoff, Felipe Pereira

THE UNIVERSITY OF TEXAS AT DALLAS
School of Natural Sciences and Mathematics

14 March 2019

Outline

Exploration Seismology

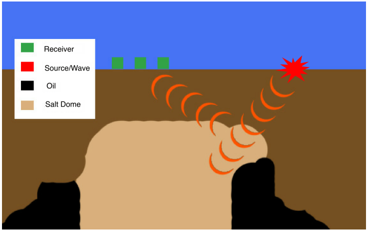


Figure: The reflection seismology process. Waves are generated at the source and reflect off the interfaces between different materials.

- In exploration seismology, seismic waves (acoustic or elastic) can be used to image the subsurface of the earth.
- In a typical seismic experiment:
  1. A source creates a disturbance in the form of a wave.
  2. This wave travels through the earth and reflects off of material property interfaces.
  3. Seismometers on the surface of the earth or in wells record the returning wave.
- This recorded seismic data can be used to image the earth's subsurface.
- In velocity inversion, the result is a map of wavespeed that can be used to determine lithology.

Why use Markov chain Monte Carlo?

- A deterministic approach to waveform inversion results in a single model of the desired parameter. Constructing uncertainty information requires many assumptions about a single model, even Bayesian formulations of the inverse problem.
- A stochastic approach allows us to characterize and quantify uncertainty with fewer assumptions.
- Markov Chain Monte Carlo (MCMC) allows us to sample from the posterior distribution of the model. We examine tens of thousands of possible velocity models to construct a picture of the posterior distribution.
- This allows us to avoid assumptions when constructing and analyzing the posterior distribution, which means a better characterization of the uncertainty.

- Mosegaard and Tarantola (1995) pioneered the use of Stochastic Bayesian methods in seismic inversion.

- Sambridge and Mosegaard (2002) summarized the use of Monte Carlo and MCMC algorithms in geophysical inverse problems.

- Bayesian methods have been used, for example, in seismic imaging (e.g., Ely et al. (2018)), reservoir flow (e.g., Oliver et al. (1997), Ginting et al. (2015), and hydrology (e.g., Vrugt et al. (1998)).

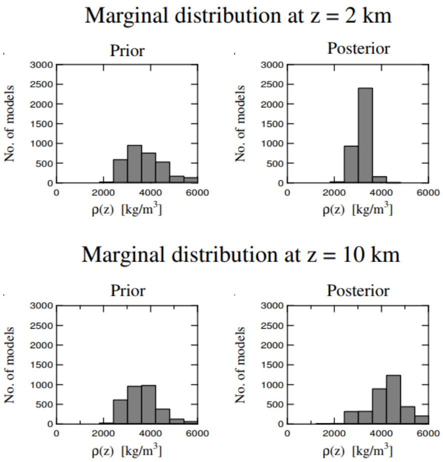


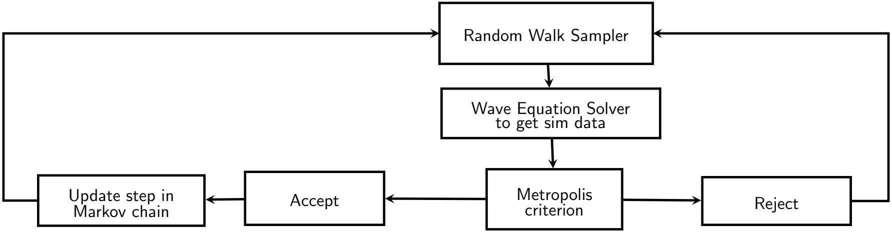
Figure: Prior and posterior distributions of mass density. Mosegaard and Tarantola (1995)

Upscaling and MCMC Velocity Inversion

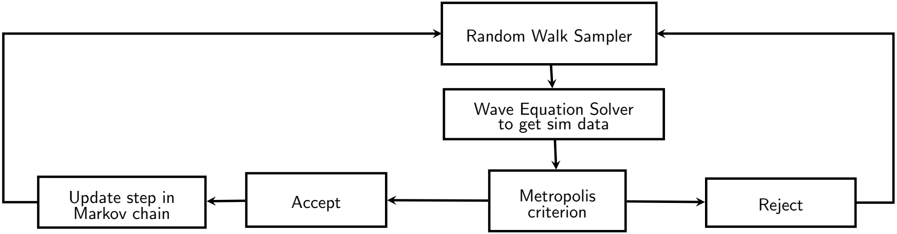
Figure: Rejected velocity models

- Problem: MCMC can take many models (tens of thousands) to converge to steady state, and each model must be run through a forward simulator to see if it is acceptable for the characterization of the posterior distribution.
- Often 90% of samples are rejected!
- Proposed solution: use upscaled solution to quickly reject samples, then simulate on the full fine grid if upscaled sample is accepted.
- This technique was first proposed by Efendiev et al. (2005) for two-phase flow.

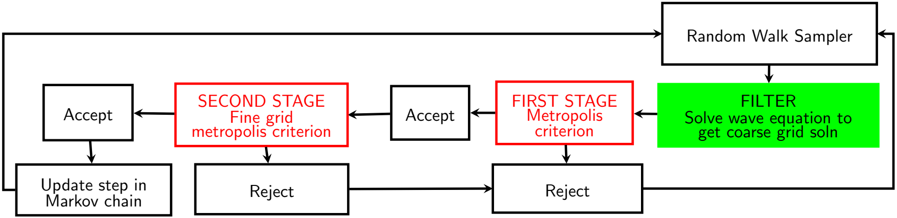
One Stage


Random Walk Sampler
Wave Equation Solver
to get sim data
Update step in
Markov chain
Accept
Metropolis
criterion
Reject

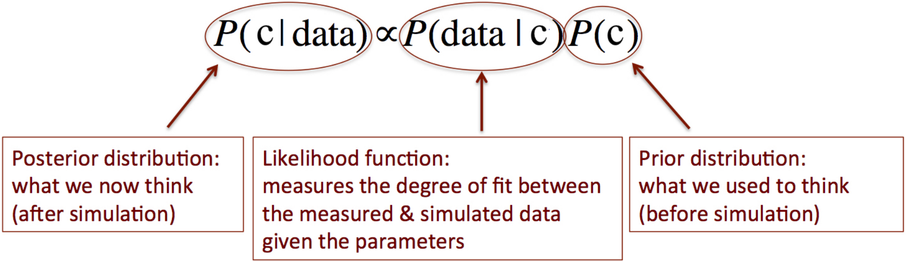
One Stage



Two Stage



According to Bayes rule,



- We assume the likelihood function has the form:

$$P(d_m|c) = \exp\left(-\frac{\|d_m - d_s\|^2}{\sigma^2}\right).$$

- The prior distribution can take many forms, e.g. uniform or Gaussian.
- However, the posterior is not necessarily Gaussian.

The Metropolis Criterion

After obtaining the simulated receiver data, we decide whether to accept or reject the proposed perturbation with the Metropolis Criterion.
Accept C with probability:

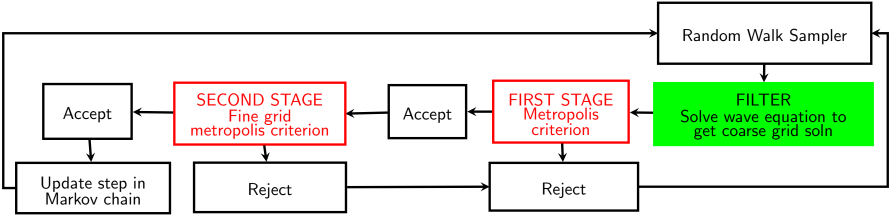
$$\rho(C_n, C) = \min\left\{1, \frac{P_F(C|d_m)q(C_n|C)}{P_F(C_n|d_m)q(C|C_n)}\right\}.$$

Where $P_F(C|d_m)$ is the posterior using the filter likelihood, C and C_n are the proposed and last accepted perturbation, $q(C|C_n)$ is the proposal distribution, and d_m is the measured data.

On the filter, we accept C with probability

$$\rho(C_n, C) = \min\left\{1, \frac{P(C|d_m)}{P(C_n|d_m)}\frac{P_F(C_n|d_m)}{P_F(C|d_m)}\right\}.$$

Two-Stage Markov Chain Monte Carlo

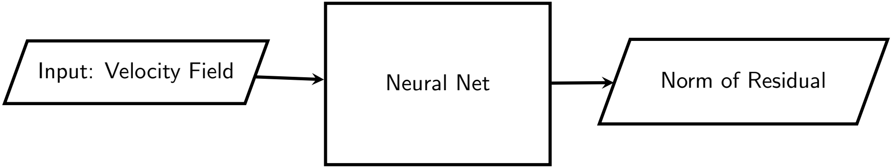


Using a Neural Net to Estimate the Likelihood Function

Likelihood function:

$$P(d_m|c) = \exp\left(-\frac{\|d_m - d_s\|^2}{\sigma^2}\right).$$

Idea: replace the expensive evaluation of $\|d_m - d_s\|^2$ with a neural net.



Advantages of Neural Nets

- Once a Neural Net is trained, evaluating a model is extremely fast (milliseconds)
- Neural Nets are capable of approximating very complex relationships
- Data for training can be generated as part of the MCMC process

Advantages and Disadvantages of Neural Nets

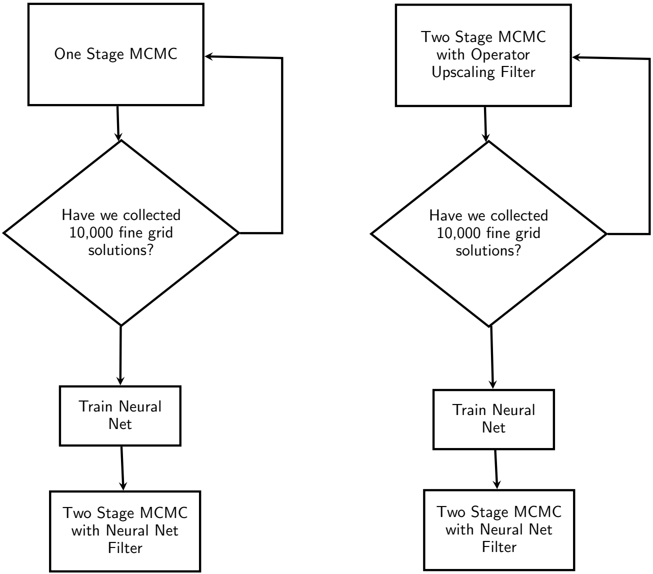
Advantages of Neural Nets

- Once a Neural Net is trained, evaluating a model is extremely fast (milliseconds)
- Neural Nets are capable of approximating very complex relationships
- Data for training can be generated as part of the MCMC process

Disadvantages of Neural Nets

- Where's the physics?
- Training data is expensive to generate
- Predictions are not always very accurate with very complex relationships
- Many knobs to twist in the Neural Net!

Training the Neural Net as Part of the MCMC Process



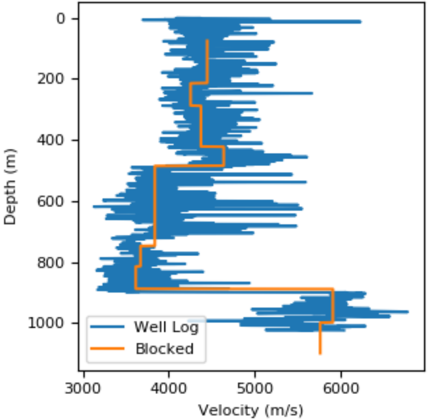


Figure: The well log (blue, courtesy of Pioneer Natural Resources) and 9-layer block (orange).

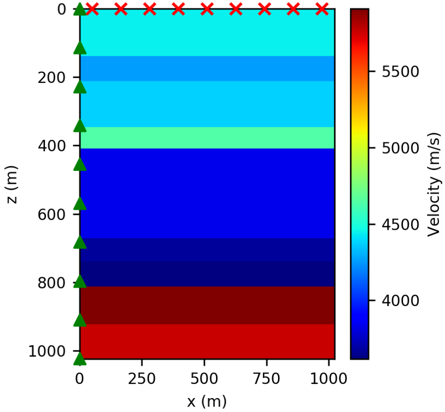


Figure: Flat Layer Experimental Setup

Training the Neural Net

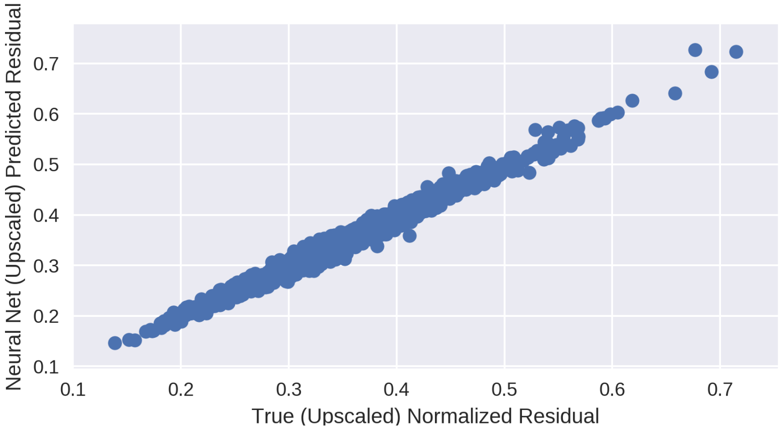


Figure: The fine grid residual norm vs neural net filter residual norm with continuous learning

Average Velocity and Relative Residuals

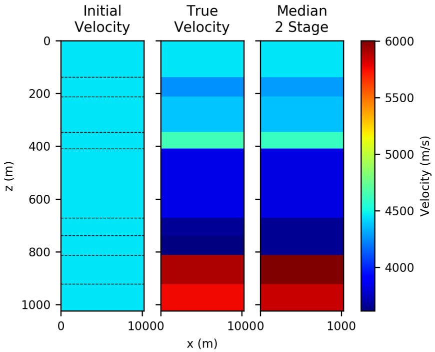


Figure: The initial, true, and median velocity fields for the neural net two stage MCMC. The dashed lines in the initial velocity picture mark the positions of the pre-set interfaces.

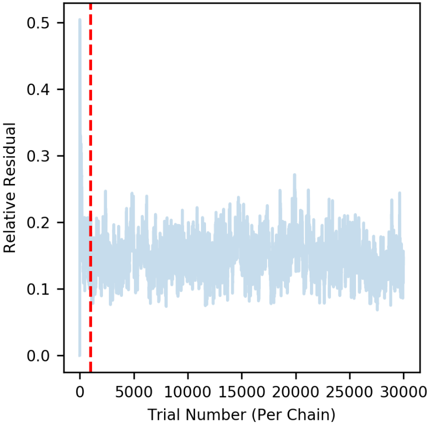


Figure: The relative residuals (blue) and burn-in cutoff (red).

Posterior Distributions

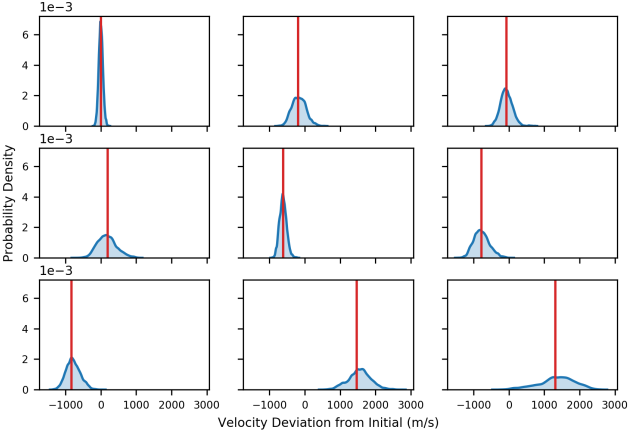


Figure: Kernel Density Estimates of the posterior distributions (blue) with the true value of the velocity (red).

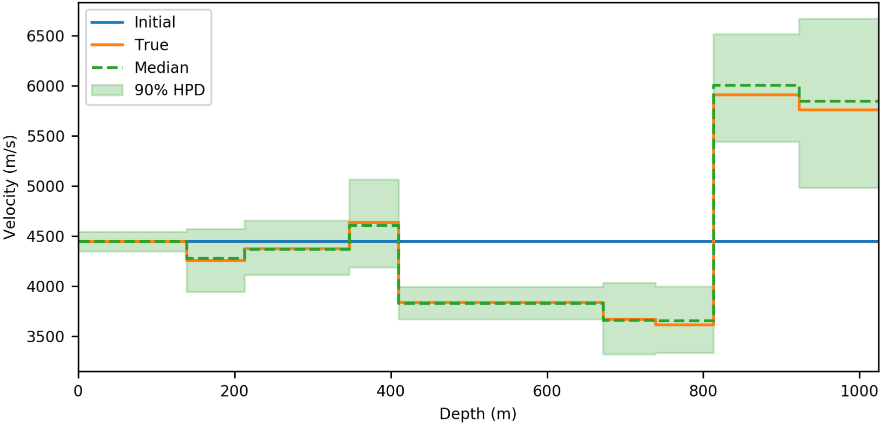


Figure: A one-dimensional slice of the velocity field in depth.

Timings

All times include generating training data and training the neural net!

One-Stage MCMC

- Time per trial: 10s
- Time per rejection: 10s
- Acceptance rate: 29%

Timings

All times include generating training data and training the neural net!

One-Stage MCMC

- Time per trial: 10s
- Time per rejection: 10s
- Acceptance rate: 29%

Two-Stage MCMC

- Time per trial: 3.5s
- Time per rejection: 1.6s
- Acceptance rate: 86%

Timings

All times include generating training data and training the neural net!

One-Stage MCMC

- Time per trial: 10s
- Time per rejection: 10s
- Acceptance rate: 29%

Two-Stage MCMC

- Time per trial: 3.5s
- Time per rejection: 1.6s
- Acceptance rate: 86%

Percent Reduction in Time Using Two Stage

- Reduction in time per trial: 65%
- Reduction in time per rejection: 84%

Conclusions

- The two-stage MCMC algorithm is an effective way to quickly reject unacceptable samples and to reduce runtime of the expensive MCMC procedure.
- A neural net is an extremely inexpensive filter (milliseconds) that can do a good job of approximating the exponent of the likelihood function.
- The training set for the neural net can be generated as part of the MCMC process.

Acknowledgments

Thank you to David Lumley and Hejun Zhu for use of the Seismology group's HPC cluster at UT Dallas.

Thank you to the Ganymede Cluster Team for use of the Ganymede cluster and their ongoing technical support.

This project was supported under the NSF Enriched Doctoral Training Program entitled "Team Training Mathematical Scientists Through Industrial Collaborations," DMS Award # 1514808.

Thank you to Pioneer Natural Resources and Robert Meek for providing industry context and data.