Efficient deterministic inversion of geophysical data constrained by Deep Generative Models to estimate subsurface parameters with defined spatial patterns

Jorge Lopez-Alvis[1,2], Eric Laloy[3], Thomas Hermans[2], Frédéric Nguyen[1]

[1]Applied Geophysics, University of Liège, Belgium

[2]Department of Geology, Ghent University, Belgium

[3]SCK-CEN (Belgian Nuclear Research Center), Belgium

AGU Fall Meeting, December 2019

LIÈGE université

GHENT UNIVERSITY

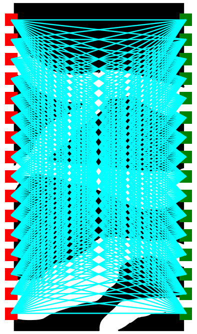
ITN Enigma

Background

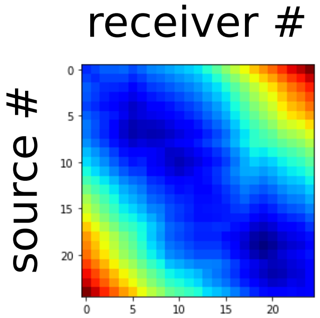
Classical imaging algorithms sometimes fail in reproducing realistic structures.

true facies

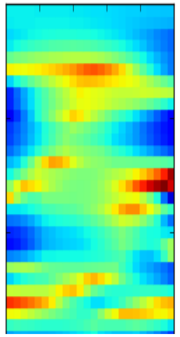
traveltime tomography



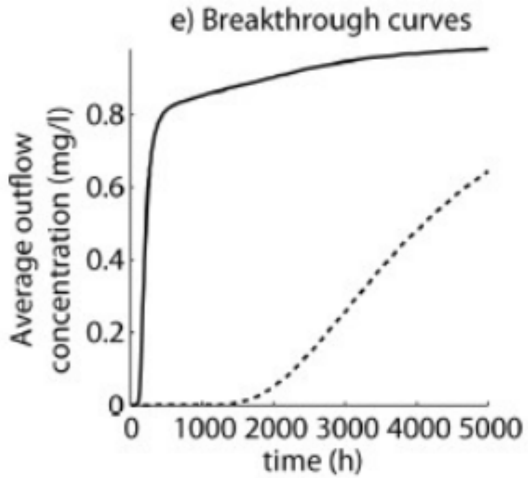
traveltime data



image



This may in turn result in wrong predictions.

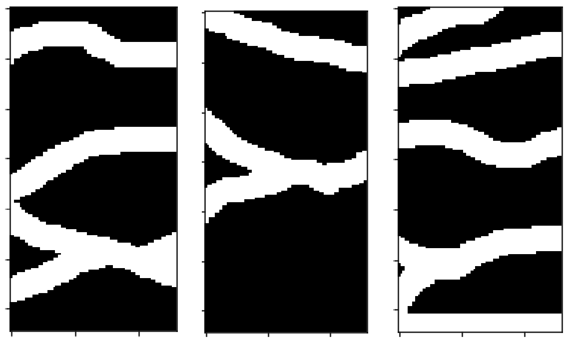


Solution: constrain the inversion to display the expected patterns (e.g. multiple-point statistics).

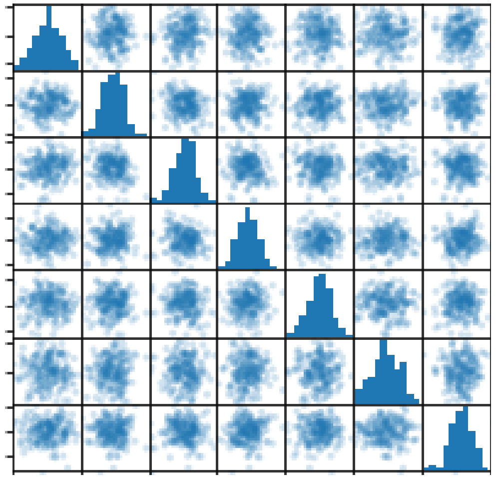
Difficulty: high-dimensional parameters usually require large number of samples (simulations) for inference.

Deep Generative Models (DGM)

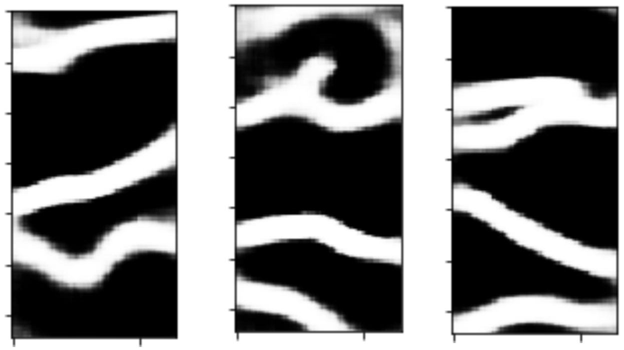
Train the DGM with many realizations (images) of expected patterns.



A latent space with lower number of dimensions is created.

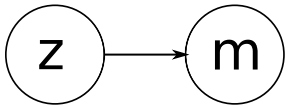


Samples with the patterns are obtained by sampling in the latent space and passing through a deep neural network.

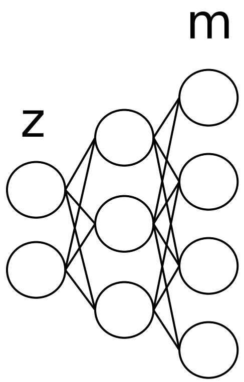


Deep Generative Models (DGM)

Implicit generative modeling



$m = f(z)$ where $f(z)$ is a deep neural network



Two big contenders: VAE and GAN

Variational Autoencoder (VAE)

Training: KL-divergence and reconstruction loss.

Advantages: easy training and high diversity.

Issues: oversmooth samples.

Generative Adversarial Network (GAN)

Training: Adversarial loss (generator vs discriminator).

Advantages: sharp samples.

Issues: difficult training and mode collapse (lack of diversity).

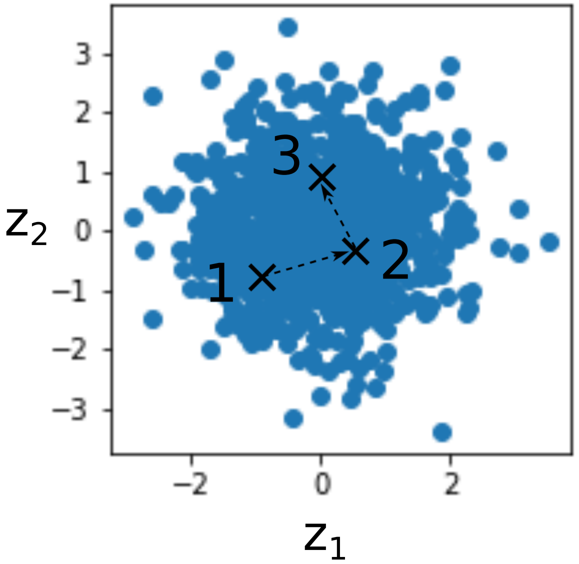
Optimization of the objective function

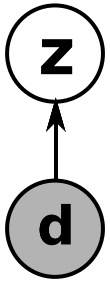
Global optimization - computationally expensive

Gradient-based optimization

With a Deep Generative Model:

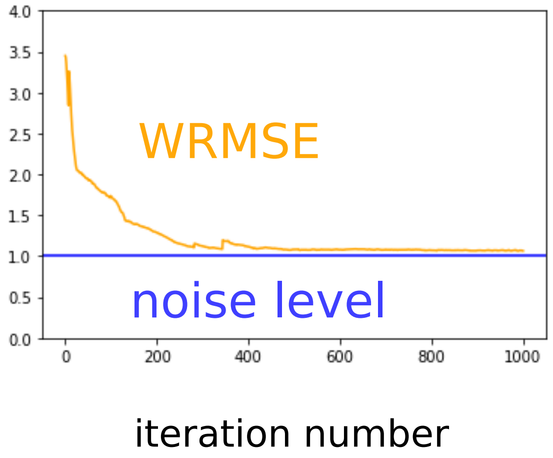
Instead of optimizing w.r.t. (gridded) parameters, do it w.r.t. latent space of DGM

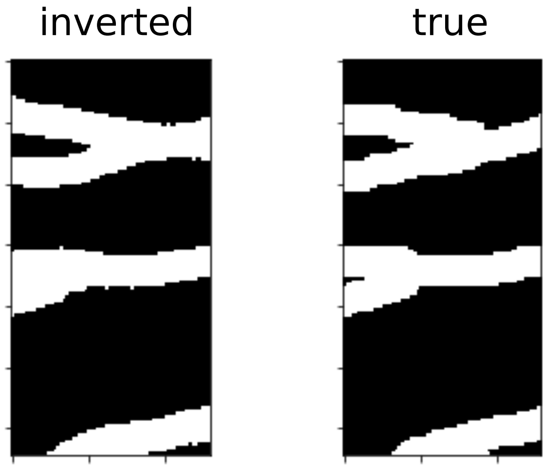




Chain rule:

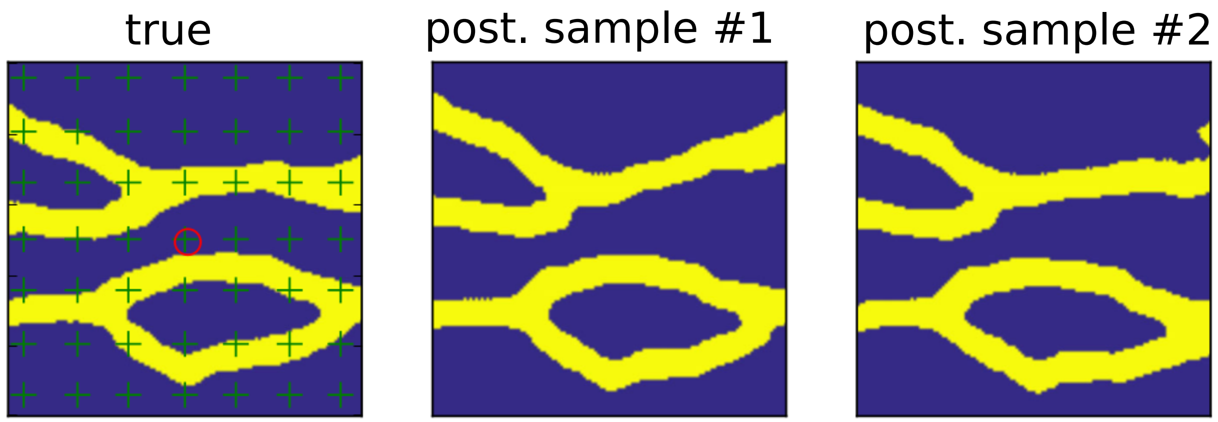
$$\frac{dS}{d\mathbf{m}}\frac{d\mathbf{m}}{d\mathbf{z}} = \frac{dS}{d\mathbf{z}}$$





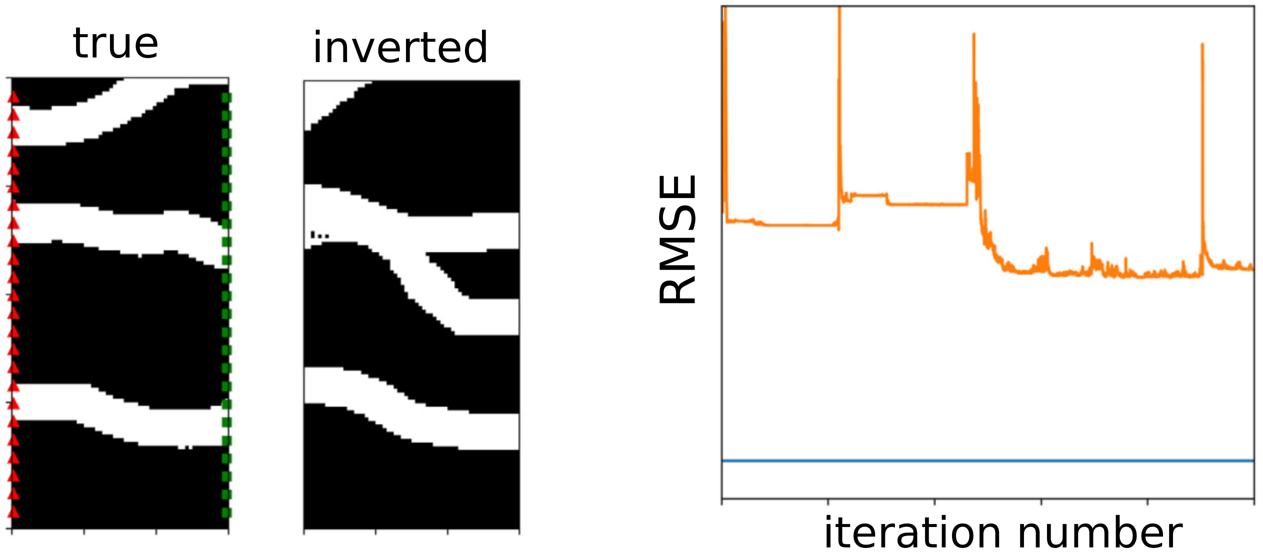
Generative Adversarial Network (GAN)

Global optimization - MCMC is working fine.



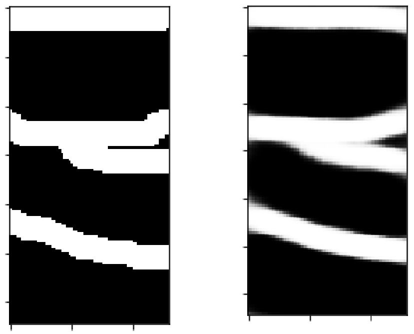
Laloy et al. 2018, Water Resources Research

Gradient-based inversion - issues with convergence to local minima.

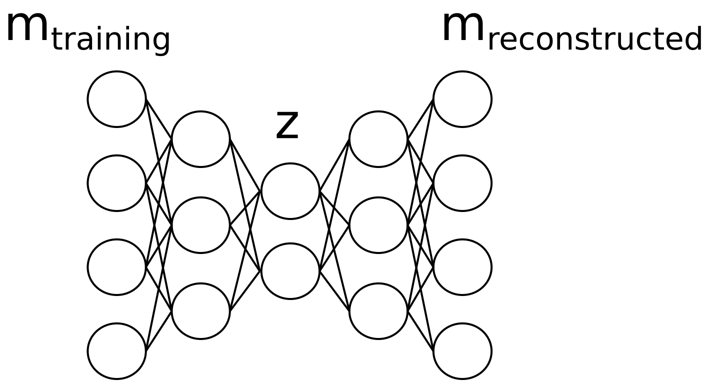


Laloy et al. 2019, Computers and Geosciences

Variational Autoencoder (VAE)

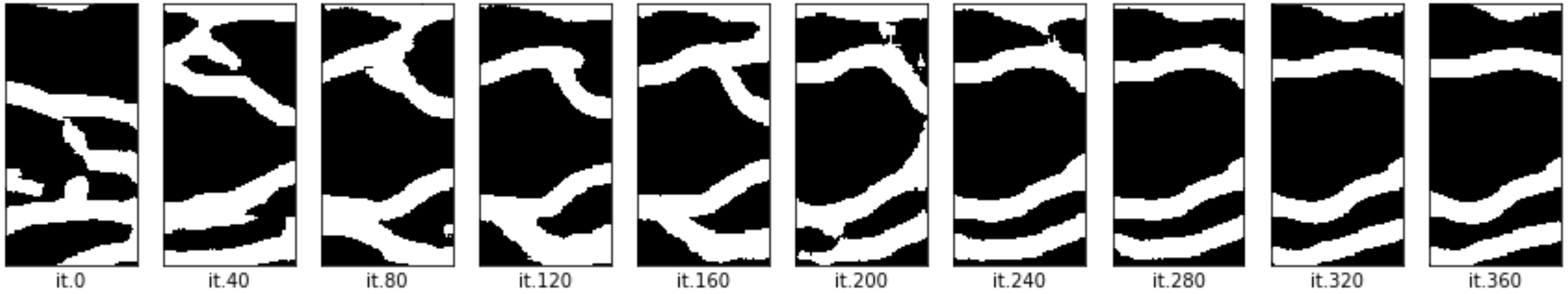


reconstruction
(not inversion)



A less strict latent space allows easier optimization?

The ability to break up channels

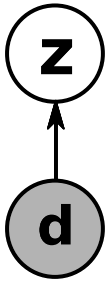


true

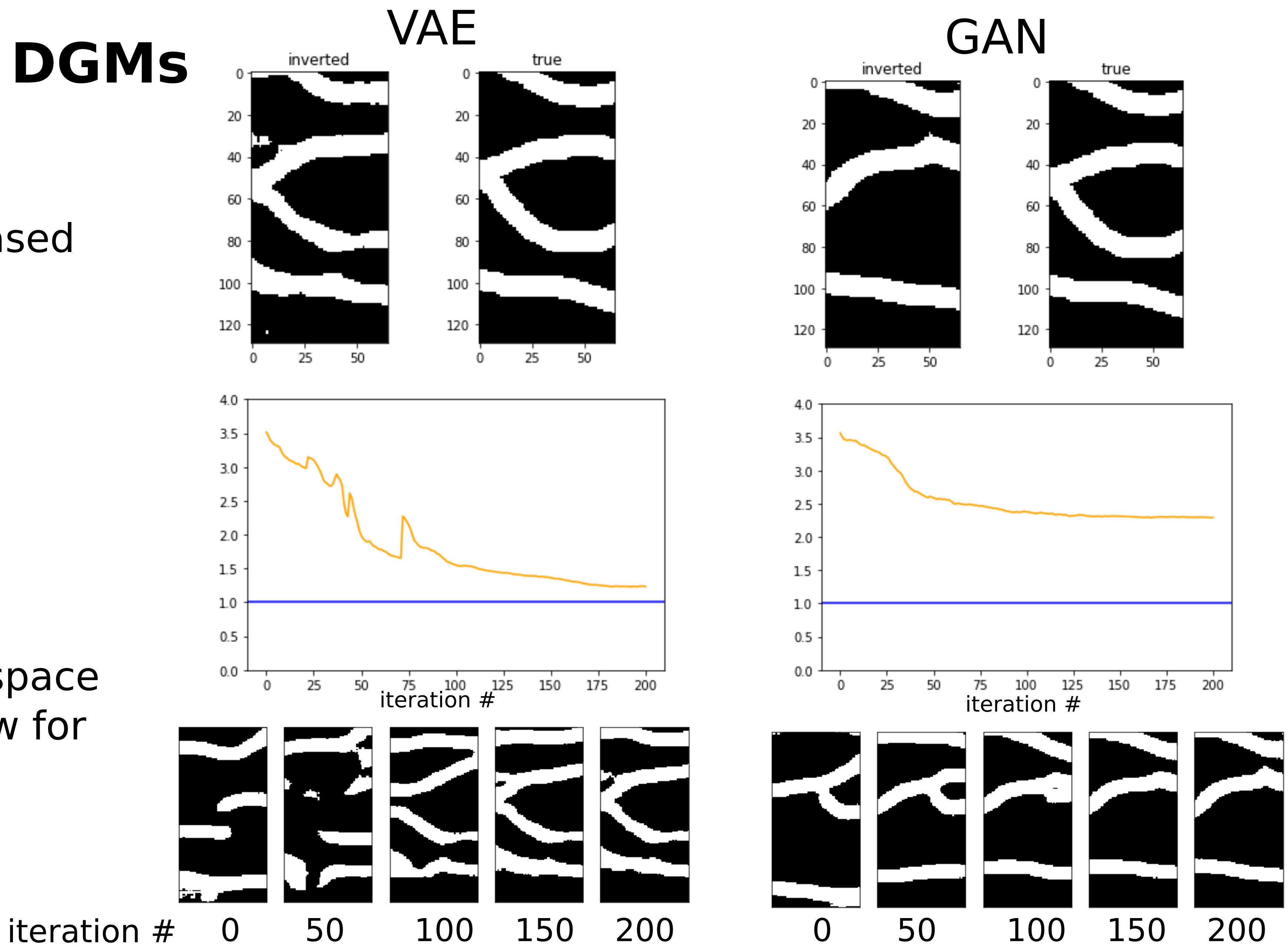
A tradeoff between the fidelity of samples and the easier optimization.

Comparison of DGMs

Inference (gradient-based inversion)



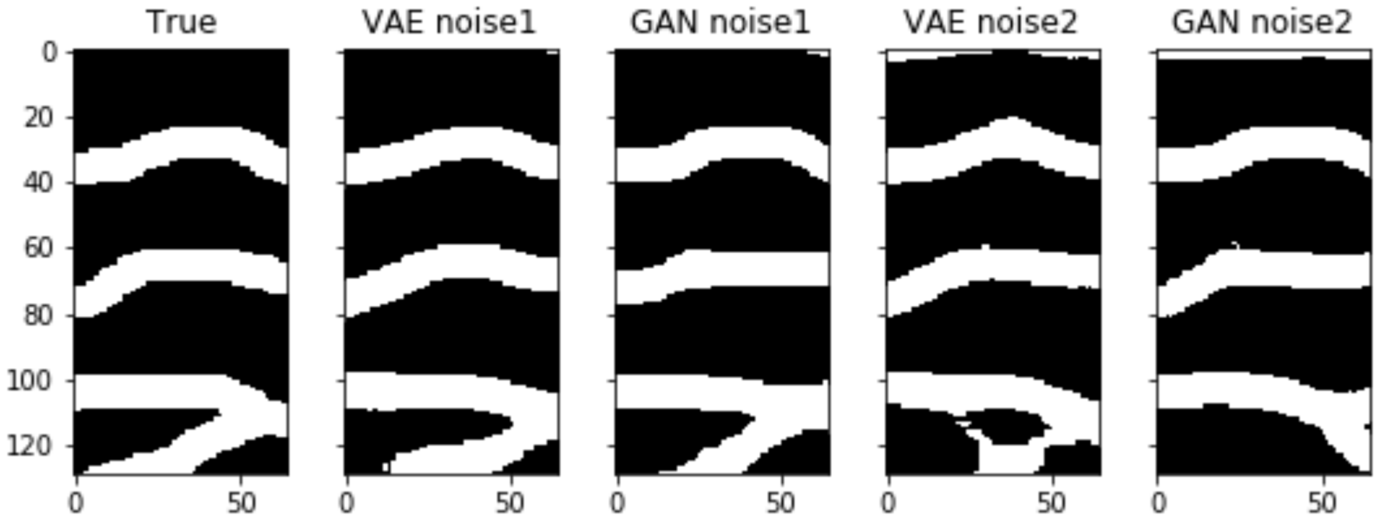
VAE induces a latent space which appears to allow for easier optimization.

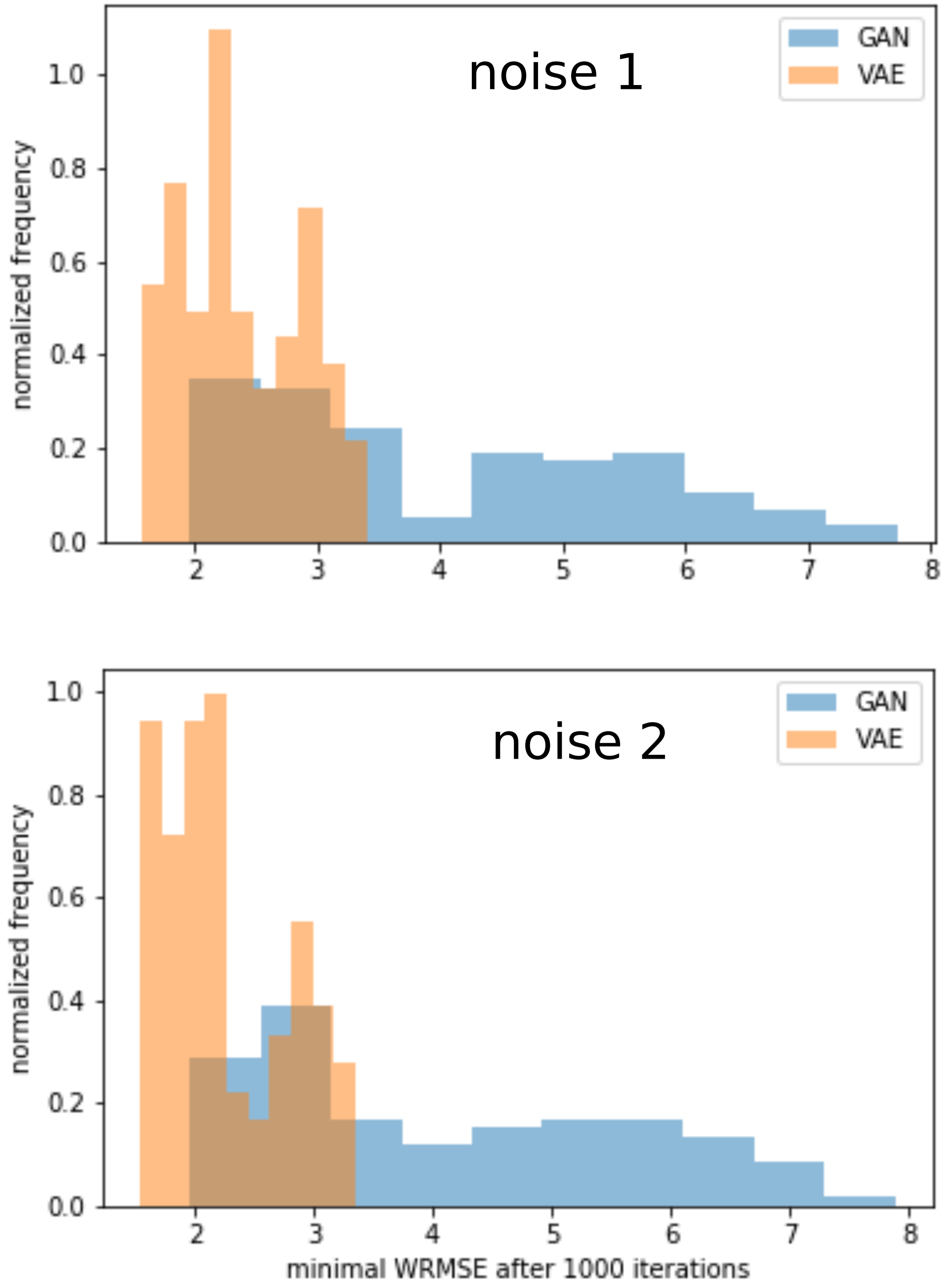


Comparison of DGMs

WRMSE comparison for gradient-based inversion with GAN and VAE (100 different initial models)

models corresponding to the lowest WRMSE (among the 100 tries):





Concluding remarks and future work

VAE latent space is easier to handle with gradient-based optimization compared to GAN latent space.

There is a tradeoff between pattern fidelity and easier optimization.

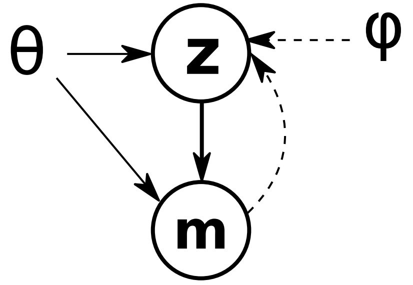
Is the method still useful for nonlinear forward operators (e.g. shortest path method)?

What is the impact of the VAE model error on the final inversion result?

VAE theory

Variational Autoencoder

(VAE)

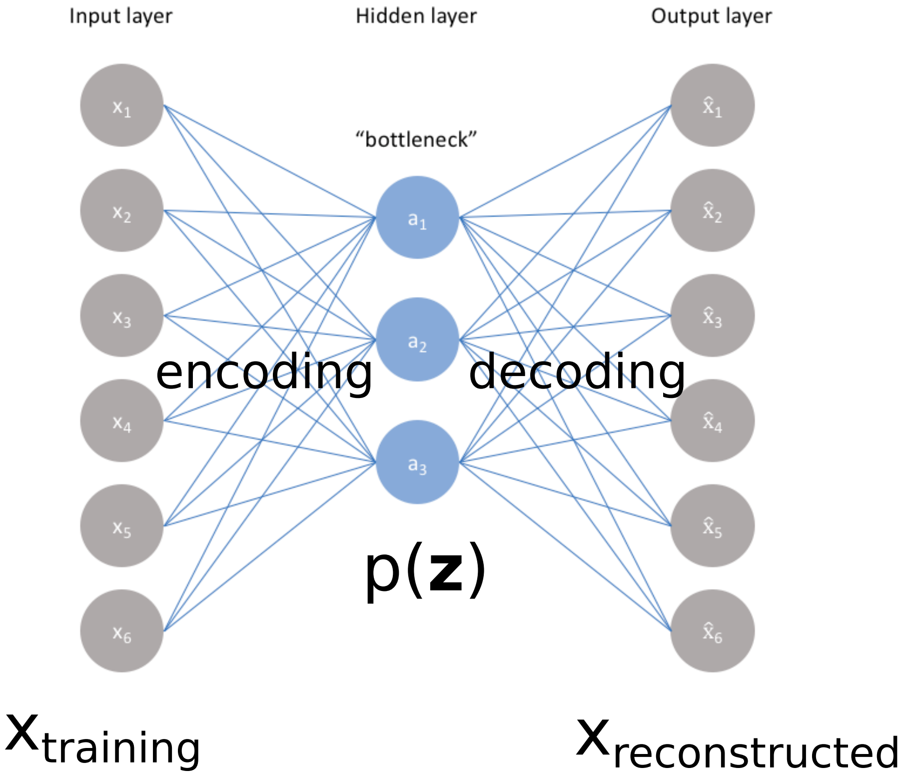


p(**z**) chosen Gaussian

p(**m**|**z**) obtained from passing p(**z**) through a DNN

Since p(**z**|**m**) is intractable, a variational q(**z**|**m**) is used whose mean is a DNN too

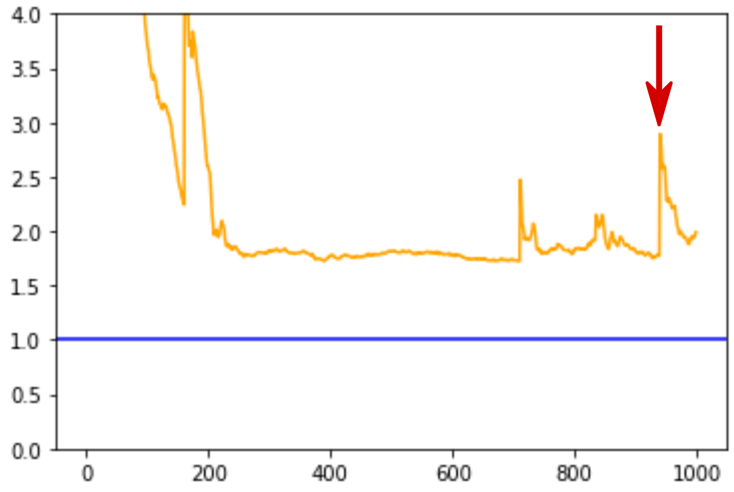
$$p(z) \equiv \mathcal{N}(0, I)$$
$$p(x|z) \equiv \mathcal{N}(f(z), cI)$$
$$q_x(z) \equiv \mathcal{N}(g(x), h(x))$$

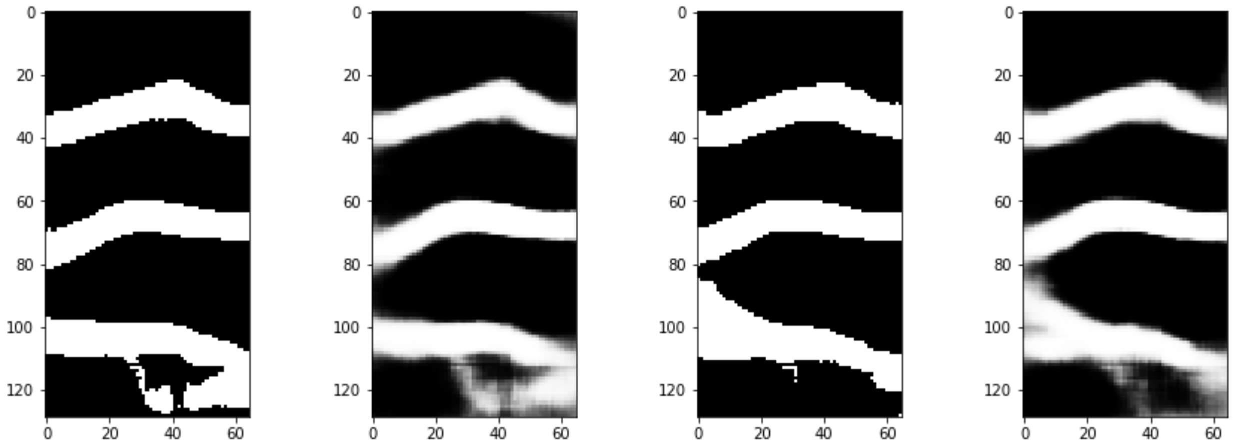


Maximize:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)}) || p_{\boldsymbol{\theta}}(\mathbf{z})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})\right]$$

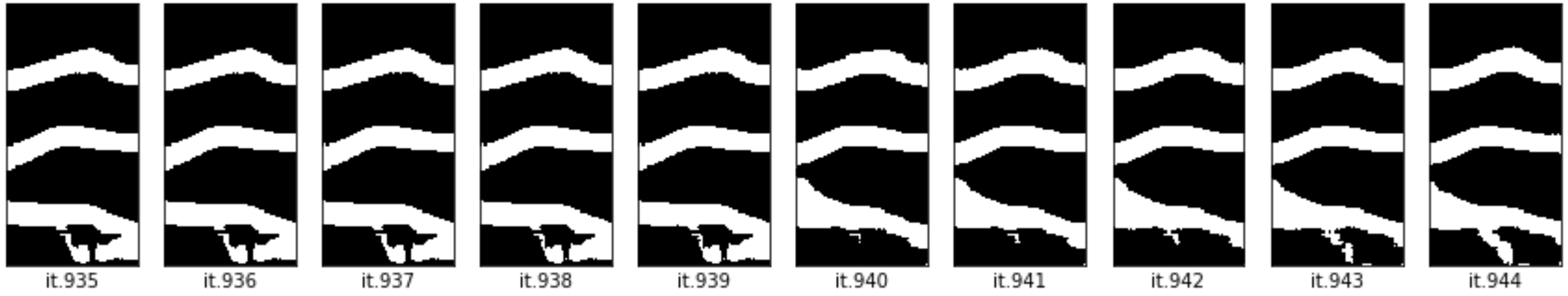
VAE jumps





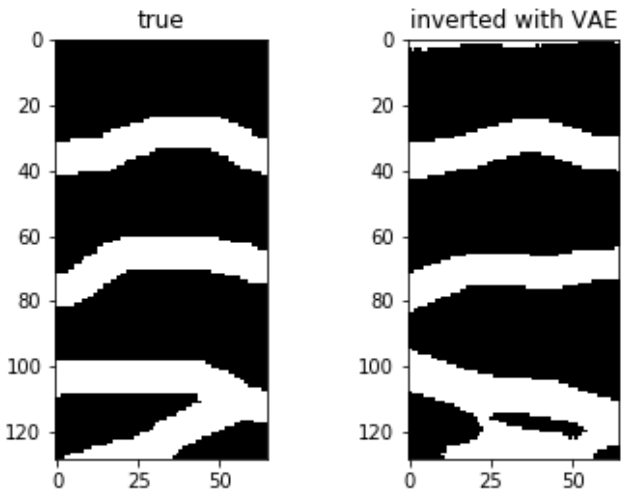
it. 939

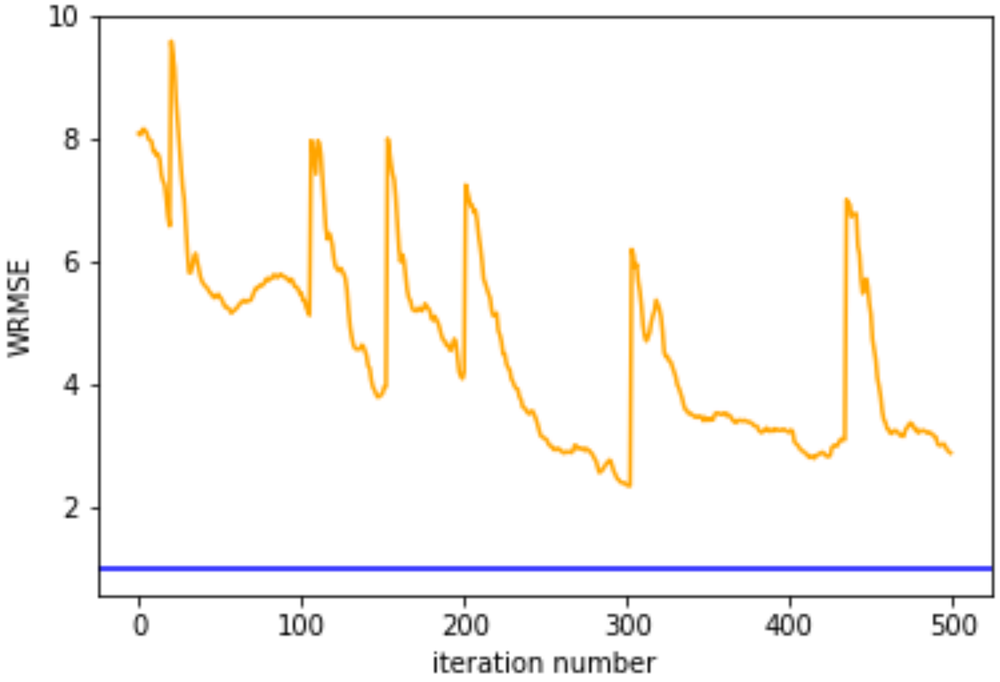
it. 940



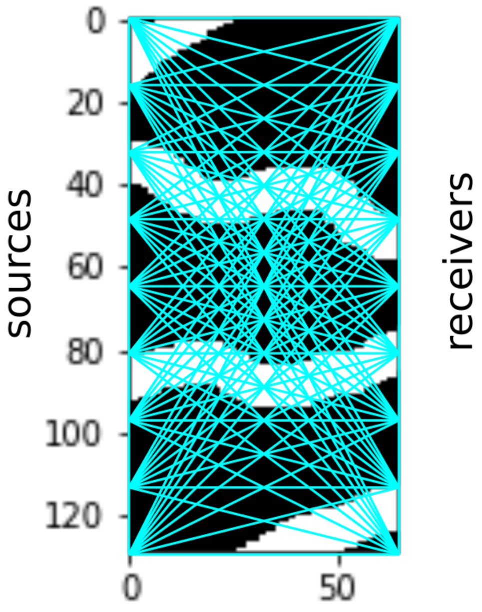
Nonlinear forward traveltime

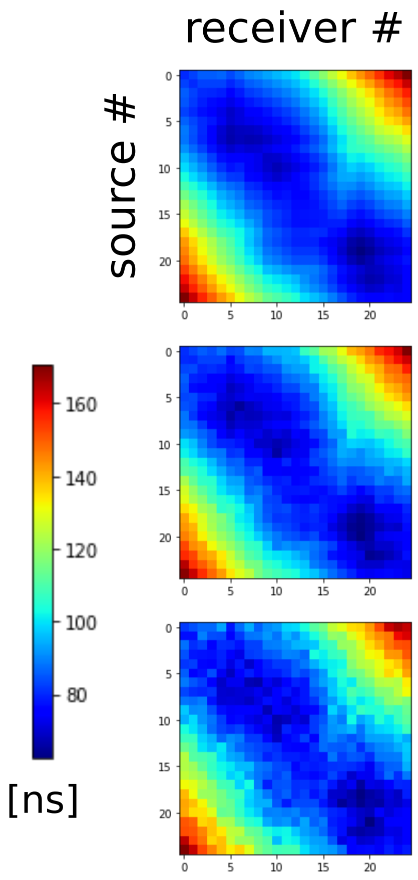
Does this work with a nonlinear forward model?





Linear vs nonlinear forward traveltime





d

d+N(0,1.4 ns)

d+N(0,3.0 ns)