Disentanglement and Improved Convexity in Geophysical Inversion with β-VAEs

John Weis
Department of Earth Sciences
University of British Columbia
jweis@eoas.ubc.ca

Justin Furlotte
Department of Mathematics
University of British Columbia
furlotte@math.ubc.ca

Grady Thompson
Department of Electrical and Computer Engineering
University of British Columbia
grady01@ece.ubc.ca

Abstract

Geophysical inversion typically requires some sort of regularization in order to produce the best geologic model of the subsurface given geophysical data. One approach to this is to use a variational autoencoder (VAE) to parameterize a lower-dimensional space where the subsurface can be represented. It has been shown that Monte-Carlo methods and gradient based inversion in this lower dimensional space allow for the recovery of geologically realistic models that honor the recovered data. In general, these methods have been applied where prior knowledge on recurring patterns in the subsurface is high, but priors on spatial locations of features is low. Here, we extend the use of VAE gradient based inversion to parameterize specific geologic settings where uncertainties on the locations and physical properties of geologic features are defined in a set of synthetic models. We show that convexity of the inversion objective function can be controlled to a degree by sampling synthetic model parameters from prior distributions on the VAE latent space.

1 Introduction

There is considerable value in being able to image the subsurface of the earth. In particular, geophysical imaging techniques can help determine the location of critical resources. *Forward modelling* is the task of, given some subsurface model m (such as a conductivity or mass distribution), computing the measurements that will be observed by above-surface instruments. The forward model, represented by $F[m]$, is entirely deterministic. The *inversion problem* is the task of, given some geophysical data measured above-surface, determining which subsurface geologic model m produced the data. Different subsurface geologic models that give rise to a set of geophysical data are not unique due to limitations of the surveys and physics. Therefore, geophysical inversion is necessary in order to obtain the best subsurface model given the geophysical data. In geophysical inversion, the subsurface model is typically discretized into cells with a given physical property (e.g. density) and data is predicted by iterative forward modeling of the discretized subsurface until the predicted data is within an error tolerance of the ground truth data. Due to the non-uniqueness of solutions to the geophysical inverse problem, in order to obtain the best physical property model, prior information regarding the geology of the subsurface must be introduced. Often times this prior information takes fairly simple forms, and recovered models, while useful for general interpretation, are not able to represent the complexity of the local geology.

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Deep generative neural networks (DGNNs) allow for the introduction of more complex geological prior information into inversion. DGNNs can be trained on a group of spatial models deemed geologically realistic for a given scenario, and a lower dimensional latent representation of the geologic models is used in the inversion. Recently, both generative adversarial networks (GANs) and VAEs have been successfully used in inversion. However, in GANs, the non-linearity of the encoding into the latent space has been shown to significantly hinder gradient based inversion. In VAEs on the other hand, recent work has showed that optimal tuning of the training parameters can reduce the non-convexity of the inversion optimization. However, these methods have primarily been used where there are high priors on the types of geologic material and general shapes and structures of materials.

To apply gradient-based inversion to geologic scenarios that might be typical of mineral exploration or large scale geologic structure delineation, it is necessary to create a synthetic data set that is representative of the expected geology. Here we show that gradient based optimization as applied to these types of problems is feasible given the VAE architecture and training set are constructed appropriately. We demonstrate this with a simple example, and then demonstrate the effective inversion in a more complex scenario.

2 DGNN Inversion and Related Work

2.1 Inversion and DGNN inversion

To avoid confusion between terms in the geophysics and machine learning communities, we refer to the subsurface physical property model as a model, and avoid the use of the term for neural networks. In order to achieve an optimal model, geophysical inversion is often framed as an optimization problem with the following objective function:

$$\phi(m) = \phi_d(m) + \alpha\phi_m(m) \tag{1}$$

where m is a discretized approximation of the subsurface of the earth that is endowed with some physical property sensitive to the geophysical survey applied. Forward modeling is then done by applying a set of operations F that constitute partial differential equations or analytical solutions depending on the problem, and data values are predicted at the locations of the observed data. The model is then iteratively altered, often through gradient based methods, until a desired value for the data misfit is reached In this paper, we will be choosing $F = G$, where G is a linear operator that computes the vertical component of the gravity response at the data locations for models cells set to fixed densities.

Here ϕ_d is the data misfit function, which punishes deviation of the forward modeled data from the acquired data. Assuming the error in each datum is distributed uniformly, it reads:

$$\phi_d = \|F[m] - d\|^2 \tag{2}$$

where d is the observed data. Due to the non-uniqueness of solutions to the inverse problem, it is necessary to introduce the regularization term $\phi_m(m)$ which places some kind of constraint on the model, such as ensuring that the model does not stray too far from a reference model. However, because of the simplicity of this form of prior information, recovered models often are not representative of the complex subsurface geology. Geophysical inversion problems also often have inherent non-convexity due to the nature of the physics. This means that a *cooling strategy* is used for the regularization parameter α, where a high value of the parameter is used to enforce convexity at the outset before being gradually decreased until a desired data misfit is reached.

DGNNs can be used to parameterize the subsurface and represent more complex geologic scenarios. In particular, VAEs have an encoder and decoder; in the context of geophysical modeling, the encoder takes training images from a dataset $\{x^i\}_{i=1}^N$ taken from the subsurface model space M and maps them to a latent space Z. The decoder dec(z) maps samples from Z back to M. Jointly training the two can give the latent space favorable properties. In the case of a VAE, two terms are used in the loss function L.

$$L = -E_{z\sim q_\phi(z \mid x^i)}[\log(p_\theta(x^i \mid z))] + \beta \cdot \mathcal{KL}(q_\phi(z \mid x^i) \mid\mid p(z)) \tag{3}$$

In VAEs, typically the likelihood $p_\theta(x^i \mid z)$ is assumed to follow a Gaussian distribution, so the first term is proportional to a reconstruction error $\sum_{i=1}^{N} \|x^i - (x^i)^r\|^2$, where samples x^i are encoded by the latent distribution $q_\phi(z \mid x^i)$, and then the decoder $\text{dec}(z)$ is used to reconstruct a copy of the original image, $(x^i)^r = \text{dec}(z)$. The second term, the ***KL-divergence***, penalizes the latent distribution $q_\phi(z \mid x^i)$ for straying too far from the prior $p(z)$. We use a slight modification of the traditional VAE called a β-VAE, which controls the KL-divergence penalty strength with a hyper-parameter β. Due to favorable computational properties, Gaussian distributions are assumed for both $q_\phi(z \mid x^i) = \mathcal{N}(\mu(x^i, \phi), \sigma^2(x^i, \phi))$ and $p(z) = \mathcal{N}(0, 1)$. The parameters θ, ϕ are learned as the weights and biases of the neural network.

Once the network is trained, rather than solve for the model m, we consider the latent representation z, in which case the inversion objective function becomes

$$\phi(z) = \|G[\text{dec}(z)] - d\|^2 + \alpha\|z\|^2 \tag{4}$$

where the gradient of the loss function can be obtained using automatic differentiation, and z is optimized until a desired misfit is reached. The regularization term keeps z close to the prescribed normal probability distribution $p(z) = \mathcal{N}(0, 1)$.

2.2 Recent Work

Research efforts have focused on the best way to implement DGNNs into inversion, and to determine the feasibility of gradient based approaches as well as extension of the method to a wider scale of geologic settings. Laloy et al. [2] used a GAN trained with convolutional nerual networks (CNN) to learn a lower dimensional representation of training models. They successfully ran inversions with Markov Chain Monte Carlo methods in the latent space. However, in gradient based inversion, which is preferable due to relatively low computational cost, convergence was shown to be highly dependent on the choice of initial model, indicating that the encoding to the latent space is highly nonlinear.

Lopes-Alvis et al. [3] showed that the mapping from the geologic model space to the latent space changes the topology of the misfit function. Due to the high amount of non-linearity of the generator and a rough approximation of the distribution of the latent space, extreme non-convexity can be introduced into the inversion, which hinders gradient based methods. However, they showed that if appropriately constructed, a VAE can be used to map from the model space to a latent space and only introduce a small amount of non-convexity into the inversion optimization. This is achieved by altering the β hyper-parameter in the KL-divergence term during training, where there is often some trade off between the sharpness of the images generated from the latent space and the convexity of the gradient based inversion. They used randomized croppings from training images to train the network, and had a high prior information on material properties and general structure consistent with shallow geophysical surveys. However, their work was not applied to typical large scale geologic scenarios, where priors on location of general structure could be stronger.

McAliley et. al [4] used a conditional VAE to preform an inversion that would be encountered in a typical large scale geologic scenario. They showed that priors could be placed on spatial locations and general geologic structures using a synthetic set of geologic models to train the VAE. However, the VAE was conditioned with data, meaning each training example had forward modeled data attached as input. While this approach allowed them to quickly sample geologic models from the posterior once the CVAE was trained, the training of network would be cost prohibitive in comparison to gradient based inversion.

3 Methods

3.1 Gravity Inversion

In order to test the effects of the synthetic data sets on the convexity of the inversion, we implemented a 2D gravity inversion using Pytorch [5]. We pulled the linear gravity operator G from the open source geophysical inversion software Simpeg [1], and converted it to Pytorch to run the inverison. To minimize the objective (4), we applied gradient descent with a modified line-search. We started with a higher value of α, and when the optimization stalled we dropped the value of α until we could

come close to the data misfit, or we hit a minimum value prescribed to α. Because the optimization was mildly convex, when it stalled, we allowed a larger jump in the gradient where the Gaussian noise was added to the gradient components. Importantly, this was not a restart, and was only meant to kick the optimization out of local mimina.

3.2 Synthetic Model Generation

For our synthetic models, we discretized the subsurface into a 65x65 cell mesh. Each cell was 10x10 meters. Due to the nonlinearity of dec(z), the misfit function ϕ_d is not necessarily convex; to test the convexity of the misfit function in the learned latent space, we first generated spherical models in a uniform background with a fixed radius and density, see figure (1).

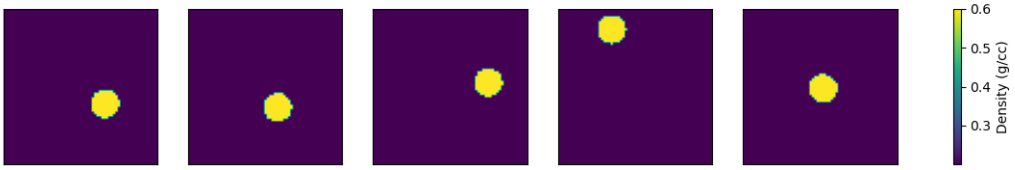

Figure 1: Examples pulled from the uniformly and normally distributed training sets for the sphere in a homogeneous background. The x and y locations were pulled from normal or uniform distributions for the normally trained VAE and the uniformly trained VAE respectively.

For the first training batch, we sampled the location of the sphere from uniform distributions in each component extending to the edges of the mesh. For the second training batch, we sampled the location of the sphere from Gaussian distributions in each component, where the edges of the mesh were three standard deviations from the mean. 10000 samples from each of these models were used as training sets for the VAE.

Additionally, we created two synthetic training sets representing more likely geologic scenarios, see figure (3). Both training sets consisted of layered earth models that are typically present in many geologic areas. For the first training set, all parameters to produce the synthetic models were pulled from normal distributions. For the second training set, all parameters were pulled from uniform distributions from within 1.75-2.25 standard deviations of the normal distribution. The layer densities and thicknesses were fixed, representing some level of geological certainty that might be available from surrounding drill-hole samples. The layer locations were pulled from either normal or uniform distributions representing geologic uncertainty in their locations. Additionally, various other parameters were pulled from normal or uniform distributions in order to simulate *geologic folding* (the wave-like characteristics seen in figure (3)). Finally, a dipping fault was added to each of the models, with the location and dip pulled from normal or uniform distributions. Although each training set had parameters pulled from these distributions, the generation of the models was a relatively complex combination of the parameters.

3.3 VAE Architecture and Training Parameters

For the implementation of the VAE architecture, we used the network described by Lopez-Alvis et al. [3] The encoder and decoder each consist of four convolutional layers and two fully connected layers. Each convolutional layer is followed by instance normalization and a leaky relu activation. The parameters we varied during the training were the β hyper-parameter in the KL misfit term, and the number of latent dimensions. The choices for these are highly dependent on the problem. For the sphere that only varied with location, we set the latent dimensionality to two. For the more complicated geologic models, following the work of Lopez-Alvis et al. [3], we selected β and the dimensions of Z based on the quality of the training sample reconstruction and the quality of randomly generated models in the latent space. A value of 200 was chosen for β for the VAE trained on the normally distributed training set. To match the reconstruction error, a value of $\beta = 150$ was selected for the VAE trained on the uniformly distributed training set. The value of the KL divergence of the uniform VAE was 1.23 times that of the normal VAE. We trained all of the VAEs for 50 epochs.

4 Results

4.1 Simple Sphere Convexity Test

Both of the uniformly drawn and normally drawn sphere locations reproduced adequate samples in the latent space in terms of the quality of the samples and the variety of the samples. However, this is not an adequate criteria for inversion, as it does not indicate the convexity of the optimization. For each sphere latent space, we calculated the loss of our objective function using a sample chosen from the synthetic set. The results are shown in figure (2) below.

For the VAE trained on uniformly drawn spheres, there is significant non-convexity of the objective function, while for the VAE trained on normally distributed spheres, the function is convex. This is evidence that training the VAE on data drawn from a Gaussian distribution is significant. This is because the KL-divergence term in the training drives the learned latent space towards a Gaussian distribution. In this case, pulling the sample locations from normal distributions forced the VAE to learn an orthogonal basis for the latent space with respect to the sphere locations. Thus, the choice of Gaussian synthetic models can promote *disentanglement*, a phenomenon occurring in β-VAEs in which a single latent component controls only one attribute of the decoded data. While complete disentanglement does not mean complete convexity of the objective function, it can be a way to help promote it.

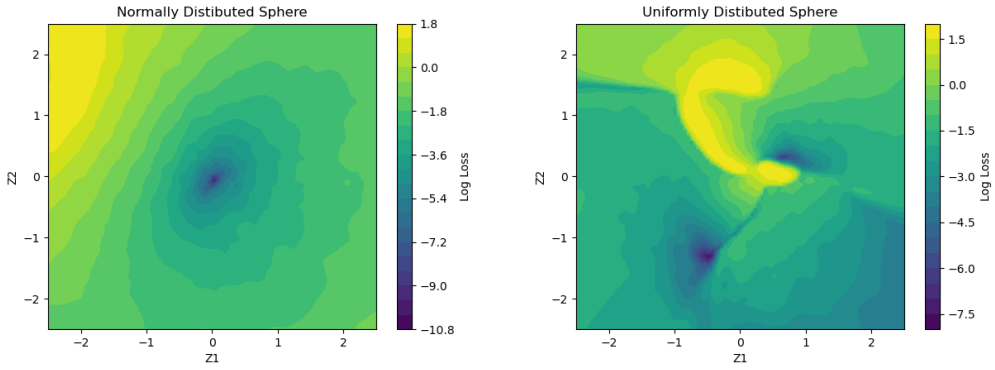

Figure 2: Both figures were generated using a sphere placed in the center. The left figure is the data misfit $\phi_d(z)$ from the VAE trained on normally distributed sphere locations, and the right figure is the data misfit from the VAE trained on uniformly distributed sphere locations.

4.2 Geologically realistic inversion

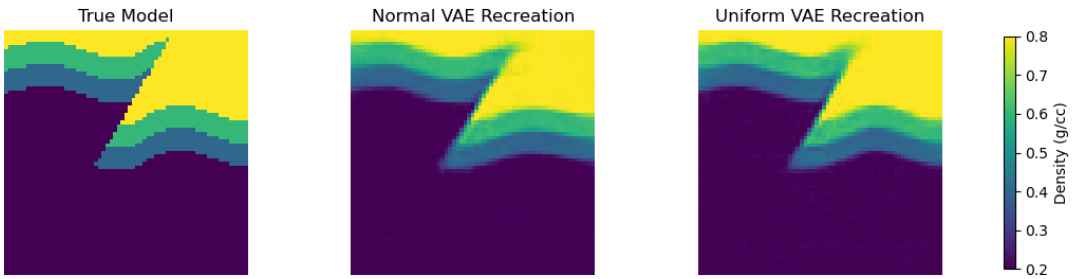

Figure 3: The true model used for testing the inversion recreated by the VAE trained on normally distributed data and the VAE trained on uniformly distributed data.

We tested our synthetic layered earth model by attempting to reproduce the model m in figure (3). No noise was added to the forward modelled data, so the recovered models were expected to be fairly close to the ground truth model. Both the VAE trained on uniformly distributed data and the VAE

trained on normally distributed data adequately reproduced the true model, as shown in figure (3). We randomly generated a set of initial models from the synthetic training set and encoded those models for the uniformly and normally trained VAE. The inversions were allowed to run for either 5000 iterations, or 5 gradient kicks with randomly added noise, and the result that had the lowest misfit was saved.

The results of the inversion and three different starting values for the inversion regularization ($\alpha = 10^3, 10^5, 10^7$) for the VAE trained on normally distributed data are shown in figure (4). All of the inversions recovered a reasonable model and reached a low data misfit. figure (6) shows the true encoded and recovered latent vectors. The sparsity of the recovered latent vectors for the high regularization shows that the VAE only needed a few components to fit the bulk of the data, indicating a fair degree of disentanglement. The recovered latent components for the normal-trained VAE are also closer in space to the true model, as indicated by the Euclidean distance from the true model.

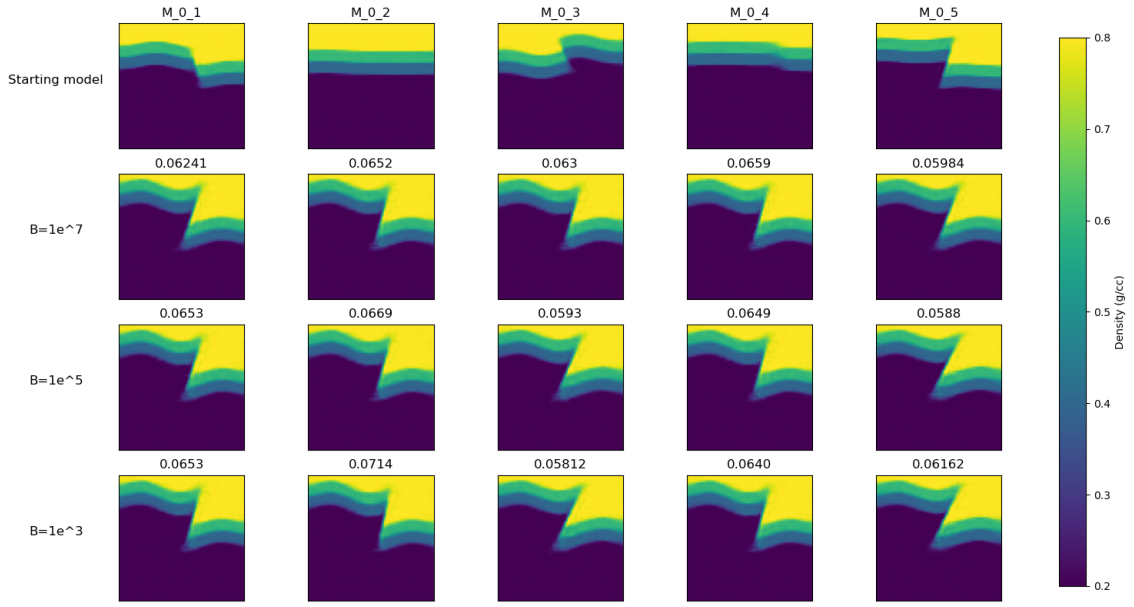

Figure 4: Recovered inversion models for five random initial models (1st row) for the VAE trained on normally distributed data. The strength of the regularization used for the inversion is labeled on the left. Each model is labeled with the recovered data misfit ϕ_d. They all achieve a reconstruction close to the desired true model from figure (3).

The results of the inversion and three different starting values for the inversion regularization ($\alpha = 10^3, 10^5, 10^7$) for the uniformly-trained VAE are shown in figure (5). While the results do place the general location of the dip of the fault, they often fail to adequately reproduce the angle of the dip. Additionally, the recovered latent vectors from the models varied more from the encoded model, as shown in figure (6). While some of the components show good agreement, many vary significantly from the true model for all starting values of the regularization, indicating that the recovered models were far in the latent space from the true model. Therefore, the allowed stochasticity in the gradient based method was not sufficient to kick the inversion out of local minima.

In general, both of the inversions preformed fairly well and were fairly similar for a range of true models. However, both also tended to get stuck in some local minima, often with similar final results. This is not surprising, as inverting for parameterized geometric structures is inherently a non-linear problem. However, for some starting models such as the example shown, the normally trained inversion significantly out-preformed the uniformly trained inversion. There are a number of possible reasons for this. When looking at the latent vector of the encoded models shown in

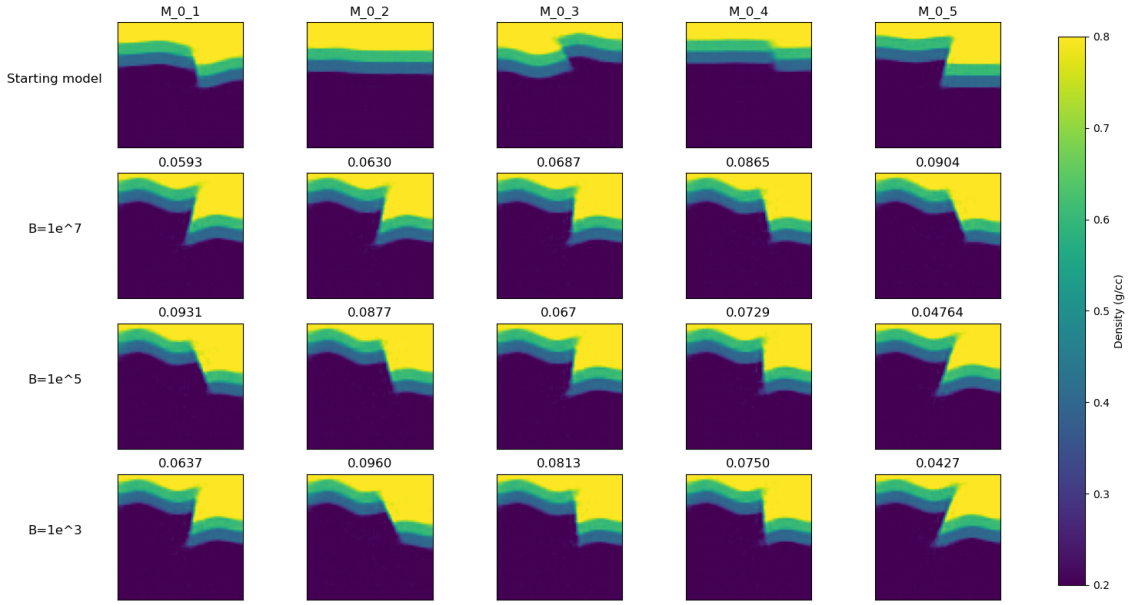

Figure 5: Recovered inversion models for five different starting models (1st row) for the VAE trained on uniformly distributed data. Many of the models failed to recover the correct orientation of the dip from figure (3).

figure (6), it is clear that the uniformly trained VAE has a component which is at the tail end of the distribution. This makes sense, as although both VAE's had similar reconstruction error, the VAE trained on the uniformly distributed samples had a higher KL Divergence. Therefore, it warped the Gaussian distribution to fit the training set. This means that the latent space was poorly regularized for the purpose of inversion, and a cooling strategy would not necessarily work. The overall difference between the true encoded and recovered latent vectors was also higher for the uniformly trained VAE, meaning that randomization methods to break out of minima are likely to be more expensive.

A second reason for the improved performance is that the level of disentanglement for the normally trained VAE is better. In particular, it was able to come close to the true model by primary using a combination of only a few of the latent components. One particularly interesting disentanglement feature we achieved was the ability to alter the angle of the fault in the inversion without significantly effecting other features of the recovered model. For gravity data that dies off as a function of r^2, this can be quite significant, as slight changes to the upper layer of the model can significantly change the data. If the latent vector components are entangled in the sense that altering the fault angle also changes the amplitude or phase of the geologic folding, the inversion may be prone to getting stuck in local minima. To test this, we built a slider applet that would update the model based on changing individual latent vector components. We picked a model with no faulting and vertically centered dip to test the disentaglement. We found a latent vector component that best changed only the dip, and altered its value from -3 to 3. The results are shown in figure (7). The normally trained VAE is able to dramatically change the angle of the fault without significantly altering other attributes, such as the waviness of the model. The uniformly trained VAE on the other hand is not able to change the dip significantly without altering the curvature of the layered earth or the locations of the layers.

A last possible reason is the particularities of the training set, and where we pulled the distributions from. All samples from the uniformly-trained VAE were pulled from 1.75 to 2.25 standard deviations of the set normal distributions. A typical sample from the normally-trained VAE would have less overall variability, but the overall training set would have included more examples of extreme individual features (sampled from the tail of the Gaussian). The typical sample from the uniform

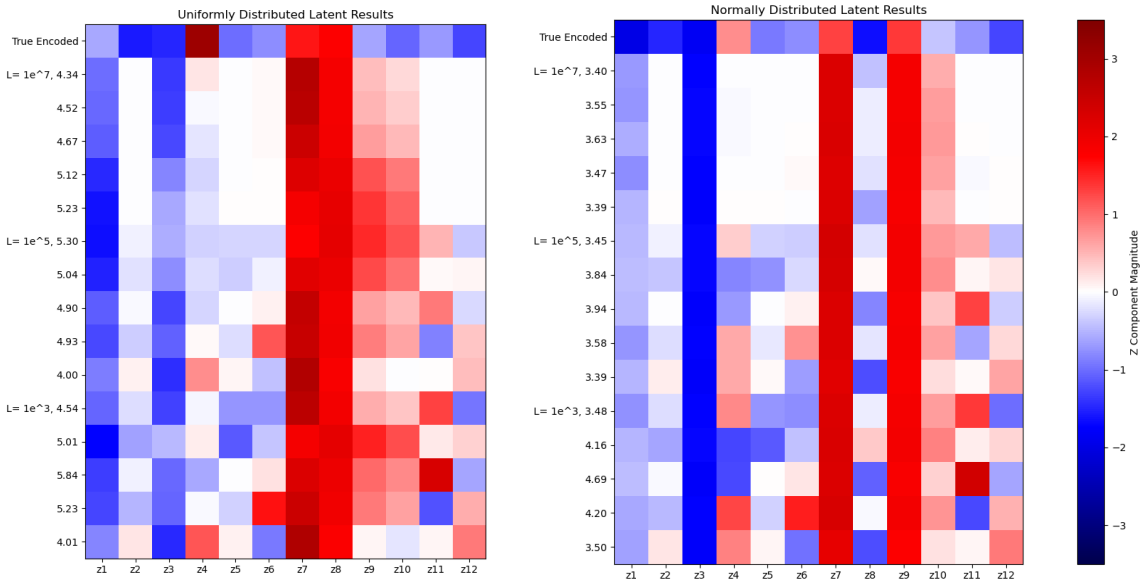

Figure 6: The true (first row) and recovered latent vectors for five different starting models and three different regularization levels. The Euclidean distance between the recovered and true encoded models are shown to the left of each recovered vector.

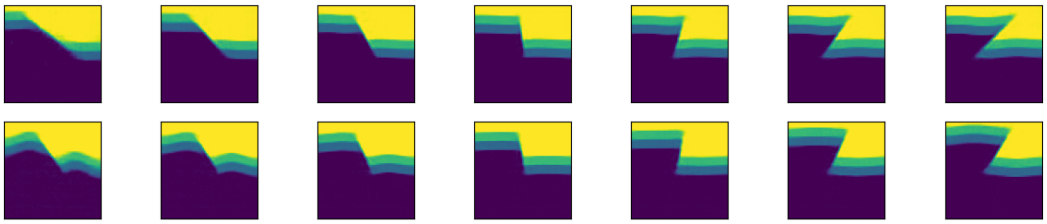

Figure 7: Results of altering one latent vector component for the normally-trained VAE (top row) and the uniformally-trained VAE (bottom row). Disentanglement is strong for the normally-trained VAE.

distribution was more diverse, but did not have any factors pulled from far outside the distribution. Therefore, the training of the uniformly-trained VAE could have been hampered because it had to fit a more diverse model on average.

5 Discussion and Future Work

In this paper we addressed how to improve convexity of geophysical inversion when inverting in the latent space of a β-VAE. In particular, we focused on how a synthetic data set could be created to introduce only moderate non-convexity into the inversion. The two dimensional sphere example shows that it is possible to enforce orthogonality of the latent components through sampling the training set from a distribution of similar form to the prior of the latent space. The geologically realistic training example admittedly had many confounding factors by nature, but results do show that using a VAE trained on normally distributed data tends to enforce disentanglement and convexity of the objective function. However, our example case was quite simple, and there are serious questions

of whether gradient based inversion is possible in this type of geologic context as the complexity of the models increases. Additionally, it seems quite limiting to have a uni-modal Gaussian prior on the auto-encoder. In future work, we would like to explore using more complex training models, introducing uncertainties into physical properties, and testing the feasibility of introducing other types of priors in gradient-based inversion.

Acknowledgments and Disclosure of Funding

The authors would like to acknowledge Andy McCaliley and Jorge Lopez-Alvis for being willing to discuss their work and share their thoughts on the directions of this research topic. Additionally, thanks to Jorge Lopez-Alvis for sharing the code for his research work.

References

[1] Cockett, Rowan, Seogi Kang, Lindsey J. Heagy, Adam Pidlisecky, and Douglas W. Oldenburg (2015) SimPEG: An Open Source Framework for Simulation and Gradient Based Parameter Estimation in Geophysical Applications." *Computers and Geosciences*, doi:10.1016/j.cageo.2015.09.015.

[2] Eric Laloy, Niklas Linde, Cyprien Ruffino, Romain Hérault, Gilles Gasso, Diederik Jacques, (2019) Gradient-based deterministic inversion of geophysical data with generative adversarial networks: Is it feasible?, *Computers Geosciences*, https://doi.org/10.1016/j.cageo.2019.104333.

[3] Lopez-Alvis, Jorge, Laloy Eric, Nguyen, Frederic, Hermans, Thomas. (2021). Deep generative models in inversion: The impact of the generator's nonlinearity and development of a new approach based on a variational autoencoder. Geosciences. 152. 104762. 10.1016/j.cageo.2021.104762.

[4] McAliley, W.A., Li,Y. (2021) Machine learning inversion of geophysical data by a conditional variational autoencoder *Society of Exploration Geophysics* 10.1190/segam2021-3594761.1

[5] Adam Paszke et al.(2019) PyTorch: An Imperative Style, High-Performance Deep Learning Library http://arxiv.org/abs/1912.01703

[6] Lopez-Alvis, https://github.com/jlalvis/VAESGDfield