Designing Deep Inverse Models for History Matching in Reservoir Simulations

Vivek Sivaraman Narayanaswamy[*], **Jayaraman J. Thiagarajan**[†], **Rushil Anirudh**[†], **Fahim Forouzanfar**[‡], **Peer-Timo Bremer**[†], **Xiao-Hui Wu**[‡]
[*]Arizona State University, [†]Lawrence Livermore National Laboratory,
[‡]ExxonMobil Upstream Research Company

Abstract

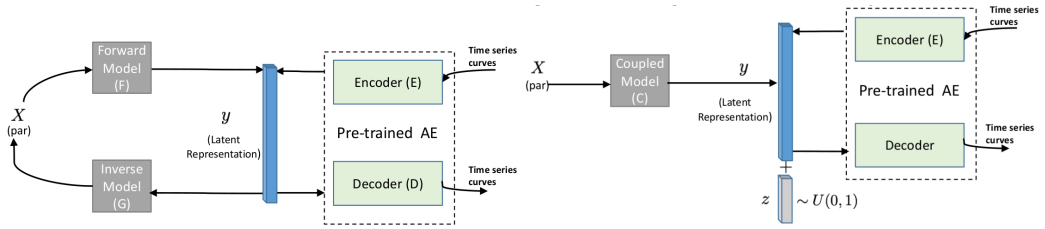
In a wide-range of applications in science and engineering, one often faces the need to learn complex mappings between independent parameters and dependent/measured quantities, i.e. the *forward* and *inverse* mappings. Building reliable inverse maps characterizing the conditional posteriors is challenging since in practice the mappings are seldom bijective. Moreover, it is challenging to incorporate scientific priors into the learning process. In this paper, we argue that enforcing self-consistency between forward and inverse models is an effective regularizer for learning predictive models in scientific applications. In particular, we develop two different strategies to enforce self-consistency, namely cyclical and coupled training methods. Using data from a reservoir model simulator, we apply the proposed approaches for history matching, which is the process of identifying the distribution of parameters that best explains the observed data. Our results show that self-consistency is highly beneficial, and both training strategies produce well calibrated inverse models.

1 Introduction

As computer simulations are becoming more reliable, they are increasingly used to interpret experimental data or design new experiments. While the details vary widely depending on the specific application, the basic goal remains to find simulation parameters whose outcome almost mimics the observed data. If successful, the resulting parameters can provide insights into the underlying processes of the observed phenomenon, which can in turn be used to plan future experiments. Mathematically, one solves an *inverse* problem of mapping the outputs of a given simulation to the corresponding inputs. In practice, the original simulator is often too expensive to support the necessary optimization and instead an approximate *forward* model, also called a surrogate, is fitted to a given simulation ensemble [1]. This creates two coupled problems: First, fitting a forward model $\mathcal{F} : \mathrm{x} \mapsto \mathrm{y}$ that approximates the original simulation by mapping the inputs x to the outputs y; and Second, deriving an inverse model $\mathcal{G} : \mathrm{y} \mapsto \mathrm{x}$. Even for complex inputs and outputs, modern machine learning techniques have proven effective for building forward models [1]. However, the inverse models are known to be more challenging since they are often *ill-posed* as $\mathcal{F}$ can be *non-bijective* with some of the input parameters having little or no influence on the results and for others, multiple distinct combinations may produce the same outputs.

In this paper, we consider the problem of history matching in reservoir models, an inverse problem aimed at estimating and calibrating the parameters of a simulator to best match the empirical observations. Since this estimation process is non-unique, determining a single history matched model that explains the observed measurements is an ill-posed problem [2]. Consequently, it is critical to estimate all possible inverse estimations to improve the reliability of a model [3]. In other words, rather than providing a single response for a given target observation y_h one must find the

Second Workshop on Machine Learning and the Physical Sciences (NeurIPS 2019), Vancouver, Canada.



(a) Cyclical Training (b) Coupled Training

Figure 1: Proposed training strategies for building deep inverse models in history matching

posterior density on all possible parameter settings, i.e. $p(\mathrm{x}|\mathrm{y} = \mathrm{y}_h)$. Since densely sampling the input space with the original simulator is infeasible, the inverse is typically fitted by using the forward model as the likelihood function and utilizing statistical learning techniques to infer the posterior.

Especially given a relatively few initial evaluations of the simulator, directly fitting a standalone inverse model $\mathcal{G}$ produces poor results. The most likely reason is that without additional regularization or any knowledge of physics informed priors, it is difficult to produce meaningful models. Hence, more recent approaches have resorted to jointly estimating the forward and inverse processes using a direct coupling scheme [3]. The core idea of our approach is to train an autoencoder to implicitly represent the relationships between individual outputs and to systematically couple the forward/inverse models for enforcing self-consistency. In particular, we consider two different strategies, namely cyclical and coupled training. In the former case, as shown in Figure 1(a), we fit two independent models for $\mathcal{F}$ and $\mathcal{G}$ that are explicitly tied together using a cyclical consistency constraint. On the other hand, the coupled training scheme shown in Figure 1(b) utilizes affine coupling to construct a single model representing both forward and inverse processes. Though the notion of cyclical consistency has appeared in several image-image and visio-linguistic [4] applications as an effective regularization strategy, its efficacy to solve inverse problems has not yet been explored. From our experiments, we find that both the proposed strategies produce accurate and well-calibrated predictions, thus evidencing the effectiveness of self-consistency in scientific problems.

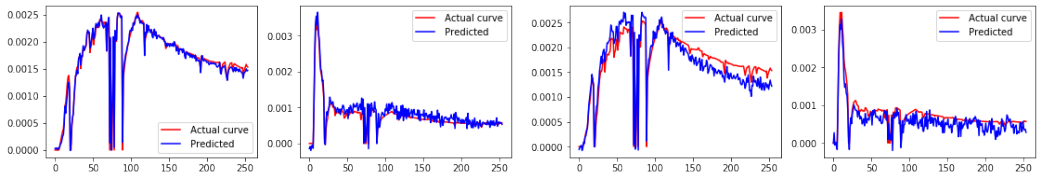
2 Proposed Approach

In order to jointly infer the forward and inverse models while enforcing self-consistency between them, we propose two strategies, namely *cyclical* and *coupled* training. Given an experiment design (e.g. Latin Hypercube) for the input parameters $\mathrm{x} \in \mathbb{R}^D$, we build a forward model $\mathcal{F} : \mathrm{x} \mapsto \mathrm{z}$, where $\mathrm{z} \in \mathbb{R}^d$ is the latent space inferred using a pre-trained encoder $\mathcal{E} : \mathrm{y} \mapsto \mathrm{z}$. Here, $\mathrm{y} \in \mathbb{R}^{T \times k}$ corresponds to a multivariate time-series output from the simulator comprised of k quantities and T time-steps. Correspondingly, the inverse $\mathcal{G} : \mathrm{z} \mapsto \mathrm{x}$ is used to recover the input parameters from a sample in the latent space. Our dataset is comprised of tuples $\{(x_i, z_i)\}, i = 1 \cdots N$, where N denotes the total number of samples.

Constructing the Latent Space. We first train a $1-$D convolutional auto-encoder on the multivariate time series observations $\{y_i\}$ corresponding to the parameters $\{x_i\}$ to obtain the d dimensional concise latent representations $\{z_i\}$. More specifically, the encoder is comprised of five 1-D convolution layers, and the decoder is comprised of 5 transposed $1-$D convolution layers followed by three $1-$D convolutions. All layers employ the ReLU activation function.

2.1 Cyclical Training

As illustrated in Fig. 1(a), this approach utilizes two independent fully-connected networks representing the *forward model* $\mathcal{F}$ and the *inverse model* $\mathcal{G}$ respectively that are tied together through a cyclical consistency constraint. Note that, the inverse $\mathcal{G}$ is designed to produce both mean (μ) and variance (ν) estimates, and is optimized with the heteroscedastic loss (Gaussian likelihood assumption). The cyclical constraint ensures that the residuals from the inversion process, $\mathrm{r} = \mathrm{x} - \mu$, are controlled by the sensitivities of the forward model $\mathcal{F}$. The optimization problem can be expressed as follows:



(a) Cyclical Training (b) Coupled Training

Figure 2: Illustration of the predictions from our forward models for an example case.

$$L_{cycle} = \lambda_1 L_1 + \lambda_2 L_2 + \lambda_3 L_3, \tag{1}$$

$$L_1 = \sum_i \|\mathcal{F}(x_i) - y_i\|_2^2, \;;\; L_2 = \sum_i \frac{\|\mu_i - x_i\|_2^2}{2\nu_i} + \frac{1}{2}\log \nu_i, \;;\; L_3 = \sum_i \|\mathcal{F}(\mu_i) - y_i\|_2^2. \tag{2}$$

In our experiments we used λ_1 = 0.5 and λ_2 = 2 and λ_3 = 0.3.

2.2 Coupled Training

The proposed coupled approach model is illustrated in Fig. 1(b). This strategy uses *affine coupling* blocks similar to [3, 5] that produce a unified forward/inverse module. Every block consists of two complementary affine layers that perform a bijective or reversible transformation on the input, wherein the transformation has a trivial inverse formulation. For example, an input p is first split into two halves p_1 and p_2 and then processed using an affine coupling block to produce outputs q_1 and q_2 respectively. The forward process is given by

$$\mathrm{q}_1 = \mathrm{p}_1 \odot \exp(s_2(\mathrm{p}_2)) + t_2(\mathrm{p}_2) \quad ; \quad \mathrm{q}_2 = \mathrm{p}_2 \odot \exp(s_1(\mathrm{q}_1)) + t_1(\mathrm{q}_1). \tag{3}$$

Correspondingly, the inverse formulation is obtained as

$$\mathrm{p}_2 = (\mathrm{q}_2 - t_1(\mathrm{q}_1)) \odot \exp(-s_1(\mathrm{q}_1)) \quad ; \quad \mathrm{p}_1 = (\mathrm{q}_1 - t_2(\mathrm{p}_2)) \odot \exp(-s_2(\mathrm{p}_2)). \tag{4}$$

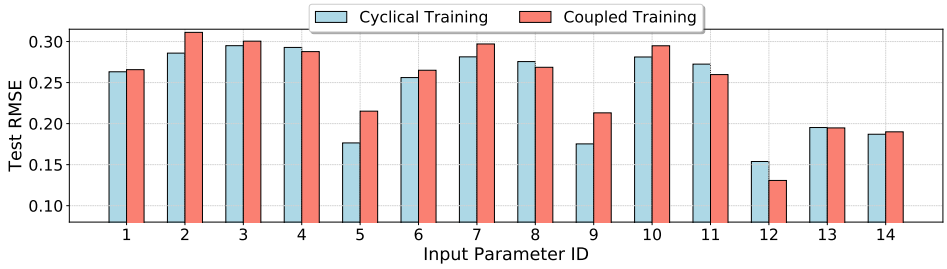
This formulation has an implicit inverse and allows the scaling function s and the translation function t to be arbitrary neural networks (implemented using fully-connected neural networks in our case). In order to allow the learned mapping to be bijective, we can optionally introduce additional noise variables in the output space, which will in turn provide additional degrees of freedom to recover spatial and multi-modal characteristics of the inputs which may have been otherwise lost during the forward transformation. More specifically, we assume that the noise variables $\mathrm{n} \in \mathbb{R}^p$ are drawn from a known distribution (e.g. uniform) and are uncorrelated to the actual output variables. In other words, $\mathcal{F} : \mathrm{x} \mapsto [\mathrm{z}, \mathrm{n}]$, and $\mathcal{G} : [\mathrm{z}, \mathrm{n}] \mapsto \mathrm{x}$. The hope is that, with a well-trained bijective mapping, one can perform Monte-Carlo sampling in the joint space of $[\mathrm{z}, \mathrm{n}]$ and recover the entire conditional distribution $p(\mathrm{x} \mid \mathrm{z})$. In order to avoid size mismatches during the affine coupling, we pad additional zeros at the input. Note that, coupled training implicitly induces cyclical consistency and hence does not require explicit constraints. In order to enforce similarity between the distributions of the original and the predicted noise variables, we use the Maximum Mean Discrepancy (MMD) [6] loss. Finally, to enable heteroscedastic loss based learning of the inverse, we propose a single-shot calibration strategy [7], wherein we generate multiple independent realizations from n and obtain empirical mean and variance estimates for the inverse, during both training and testing phases. The overall objective can be expressed as:

$$L_{coupled} = \lambda_1 L_1 + \lambda_2 L_2 + \lambda_{mmd} L_{mmd}. \tag{5}$$

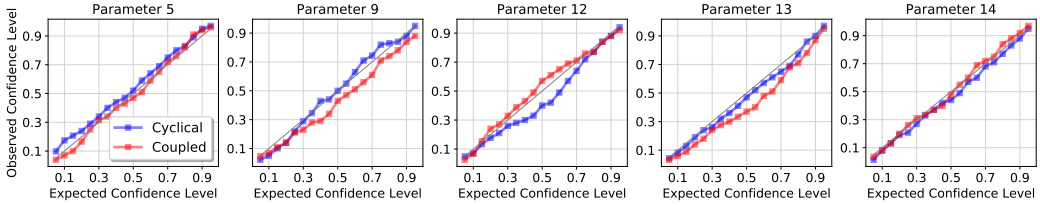
In this setup, we used λ_1 = 2 and λ_2 = 3 and λ_{mmd} = 0.5 to train the network.

3 Results

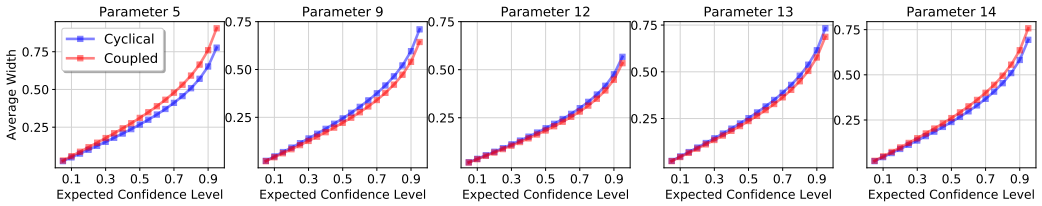
In this section, we evaluate the proposed approaches on a reservoir model simulation dataset comprised of $N = 2000$ samples of $D = 14$ dimensional input parameters along with their corresponding outputs with $k = 3$ and $T = 255$. We use randomly chosen 1900 samples for training



(a) Test RMSE of the predictions obtained using our approaches for enforcing self-consistency



(b) Calibration performance of the prediction intervals for parameters with the lowest RMSE



(c) Average widths of the prediction intervals for parameters with the lowest RMSE

Figure 3: Performance evaluation of the forward and inverse models.

and the remaining 100 examples for testing. The autoencoder was trained using the architecture described in Section 2, with the number of latent dimensions d fixed at 14. Upon training we find that both approaches produce meaningful surrogates (forward model) that can replicate the behavior of the black-box simulator with high fidelity. Figure 2 shows the predictions from both our proposed training strategies for one example case. For the coupled case, we computed the mean/variance using 20 random realizations for n. To evaluate the quality of the learned inverse mappings, which is the focus of this work, we consider the following metrics: (i) RMSE between the true parameters x and the mean estimates from the inverse model and (ii) empirical calibration where we plot the desired calibration level α against the observed calibration level (likelihood of containing the true parameter in the estimated prediction interval). For each α, the corresponding the interval $[\mu_i - \delta_i, \mu_i + \delta_i]$ can be estimated using ν as $\delta_i = z_{(1-\alpha)/2}\sigma_i$, where z indicates the $z-$score ($z_{(1-\alpha)/2} = 1.96$, for 95% calibration) and $\sigma_i = \sqrt{\nu_i}$. Finally, we also measure sharpness of the prediction intervals as

$$AW_\alpha = \frac{1}{N}\sum_{i=1}^{N} 2\sigma_i. \tag{6}$$

Note that, when an estimator is well calibrated, sharper intervals indicate that the model is highly confident about its predictions.

Our results indicate that both the proposed approaches, due to the inclusion of self-consistency, produce accurate and well-calibrated inverse mappings. In particular, we find that the cyclical training produces lower RMSE on an average (Figure 3(a)). Due to the joint inference of both $\mathcal{F}$ and $\mathcal{G}$, only parameters that are most crucial to the forward process are well-calibrated in the inverse model. As shown in Figure 3(b) and (c), both methods have low calibration errors, i.e. there is high likelihood of containing the true parameters in the estimated intervals. Interestingly, parameter 12, which had the lowest RMSE, has sharp intervals ($AW_{0.9} \approx 0.5$), indicating high model confidence, emphasizing its importance for the prediction. In summary, we find self-consistency to be highly beneficial for producing meaningful inverse models, and both cyclical and coupled training strategies are effective.

References

[1] Jacob Miller, Jayaraman J Thiagarajan, Peer-Timo Bremer, Nazish Hoda, Dave Stern, and Rick Mifflin. Data-driven metric learning for history matching. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2016.

[2] Jonathan N Carter, Pedro J Ballester, Zohreh Tavassoli, and Peter R King. Our calibrated model has poor predictive value: An example from the petroleum industry. *Reliability Engineering & System Safety*, 91(10-11):1373–1381, 2006.

[3] Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W Pellegrini, Ralf S Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

[4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[5] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[6] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

[7] Seonguk Seo, Paul Hongsuck Seo, and Bohyung Han. Learning for single-shot confidence calibration in deep neural networks through stochastic inferences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9030–9038, 2019.