DATA GENERATION: TWO PHASE FLOW

A PREPRINT

Jayjay, Tuna, Jason, Richard

2024-09-30

1 Surrogate Modeling for Which System?

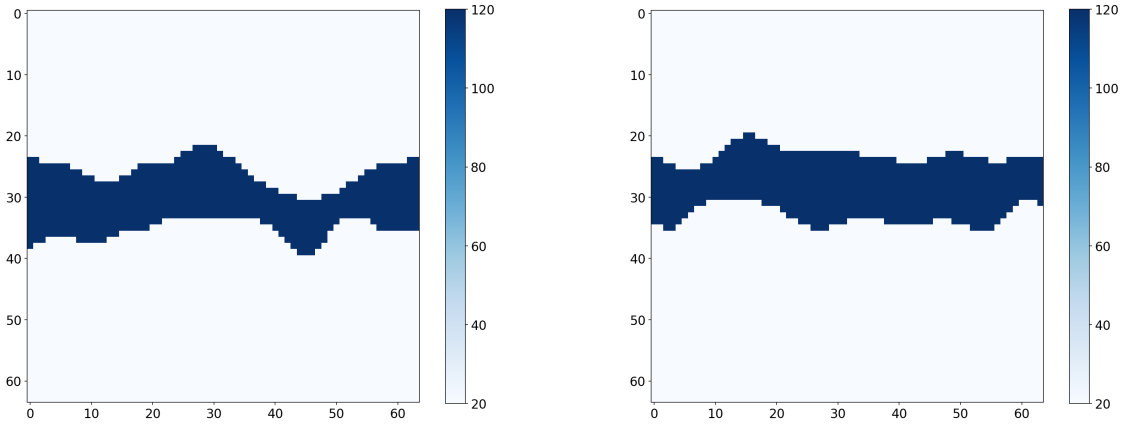
1. Simplified Geological Carbon Storage (Francis' paper)
2. Incompressible Navier Stokes

2 Twophase flow for the CO2 saturation

- We regenerate Francis' dataset, and additionally compute Fisher Information Matrix as well.
- For the purpose of validation, we currently form full Fisher Infromation Matrix and then compute eigenvector.
- Our next step will be low rank approximation or trace estimation so that we don't have to form the full matrix.

3 Dataset

Our dataset consists of 2000 pairs of $\{K, S^t(K)\}_{t=1}^8$.



(a) K0 (b) K1

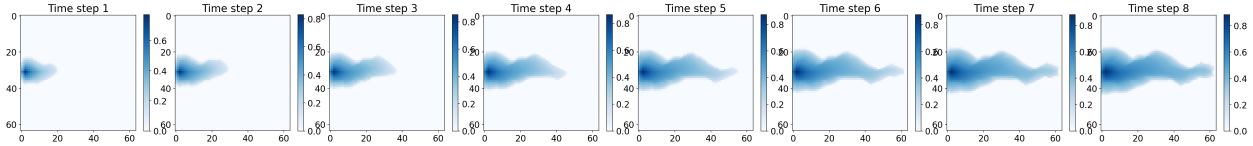
Figure 1: Example Permeability Model

4 Fisher Information Matrix

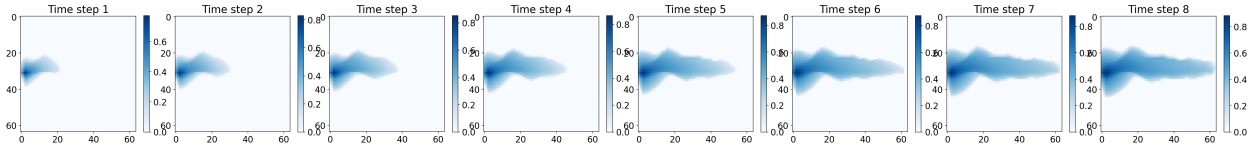
- To find the optimal number of observations, M, we visualize eigenvector and vector jacobian product.
- Given 1 pair of dataset, $\{K, S^t(K)\}_{t=1}^8$, we get a single FIM.

4.1 Computing Fisher Information Matrix for each datapoint

We consider a realistic scenario when we only have access to samples, but not distribution. When N is number of samples and $X \in \mathbb{R}^{d \times d}$, neural network model F_{nn} learns mapping from $X_i \to Y_i$. For each pair of $\{X_i, Y_i\}_{i=1}^N$, we generate $\{FIM_i\}_{i=1}^N$.



(a) Time Series of Saturation of K0



(b) Time Series of Saturation of K1

Figure 2: Example Saturation Time Series

- N : number of data points, $\{X_i, Y_i\}$
- M : number of observation, Y

$$\{X_i\}_{i=1}^{N} \sim p_X(X),\ \epsilon \sim \mathcal{N}(0, \Sigma),\ \Sigma = I$$

For a single data pair, we generate multiple observations.

$$Y_{i,J} = F(X_i) + \epsilon_{i,J}, \quad where\ \{\epsilon_{i,J}\}_{i,J=1,1}^{N,M}$$

As we assumed Gaussian, we define likelihood as following.

$$p(Y_{i,J}|X_i) = e^{-\frac{1}{2}\|Y_{i,J} - F(X_i)\|_2^2}$$

$$log\, p(Y_{i,J}|X_i) \approx \frac{1}{\Sigma}\|Y_{i,J} - F(X_i)\|_2^2$$

A FIM for a single data pair i is:

$$FIM_i = \mathbb{E}_{Y_{i,\{J\}_{i=1}^{m}} \sim p(Y_{i,J}|X_i)} \left[\left(\nabla log\, p(Y_{i,J}|X_i)\right) \left(\nabla log\, p(Y_{i,J}|X_i)\right)^T \right]$$

4.2 When Random Variable of FIM, Y, is both Saturation and Pressure

4.2.1 How does FIM change as number of observation increases?

- FIM is expectation of covariance of derivative of log likelihood. As we expected, we see clearer definition in diagonal relationship as M increases.
- We observe that as M increases, the clearer we see the boundary of the permeability, which will be more informative during training and inference. [1]

[1] Note on Learning Problem.

4.2.2 Making Sense of FIM obtained

Still, does our FIM make sense? How can we better understand what FIM is representing?

Let's look at the first row of the FIM and reshape it to [64, 64].

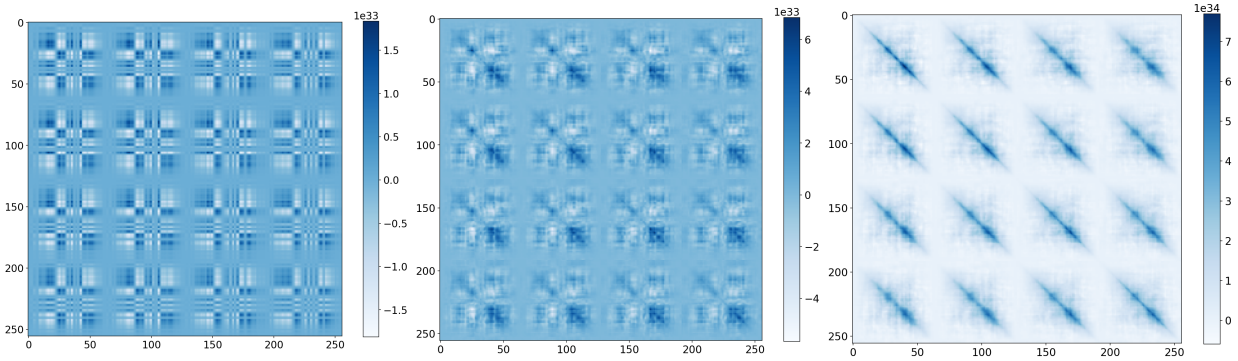
- Like we expected from the definition of FIM, we observe each plot is just different linear transformation of $\nabla logp(\{S^t\}_{t=1}^{8}|K)$
- As we will see from below, each rows in FIM is noisy version of its eigenvector.

4.2.3 How does eigenvectors of FIM look like as M increases?

4.2.3.1 $M = 1$ (Single Observation)

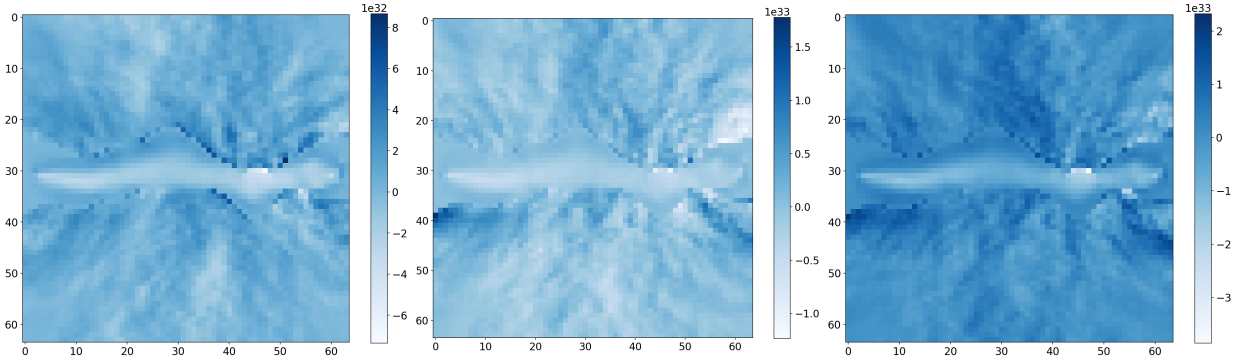
- Even when FIM is computed with single observation, we see that the largest eigenvector has the most definition in the shape of permeability. Rest of eigenvector looks more like noise.

4.2.3.2 $M = 10$



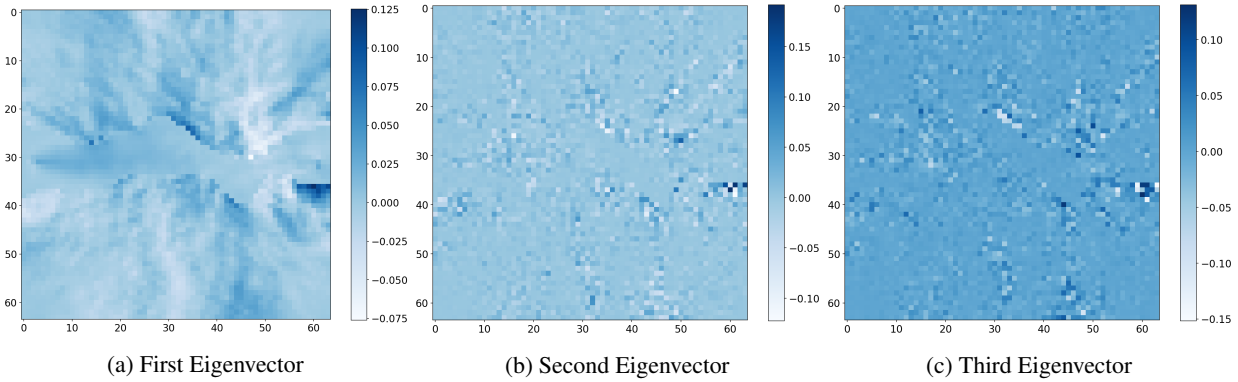
(a) M = 1 (b) M = 10 (c) M = 100

Figure 3: Change in FIM[:256, :256] of single data pair $\{K, S^t(K)\}_{t=1}^{8}$ as number of observation, M increases



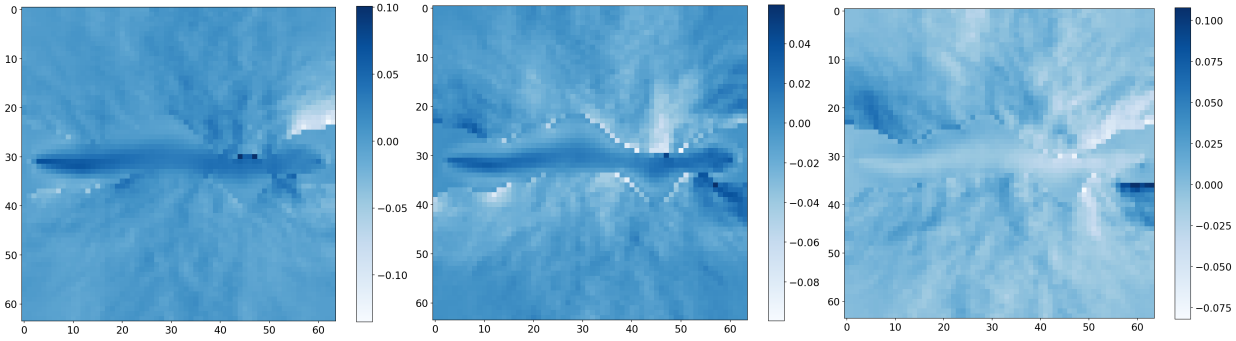
(a) FIM[0,:] (b) FIM[1,:] (c) FIM[2,:]

Figure 4: Fist, Second, and Third row in FIM



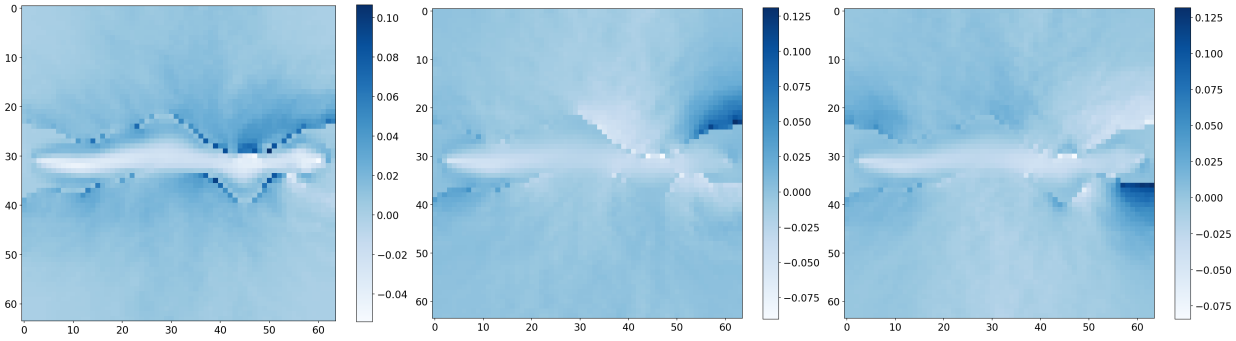
(a) First Eigenvector (b) Second Eigenvector (c) Third Eigenvector

Figure 5: First three largest eigenvector of FIM



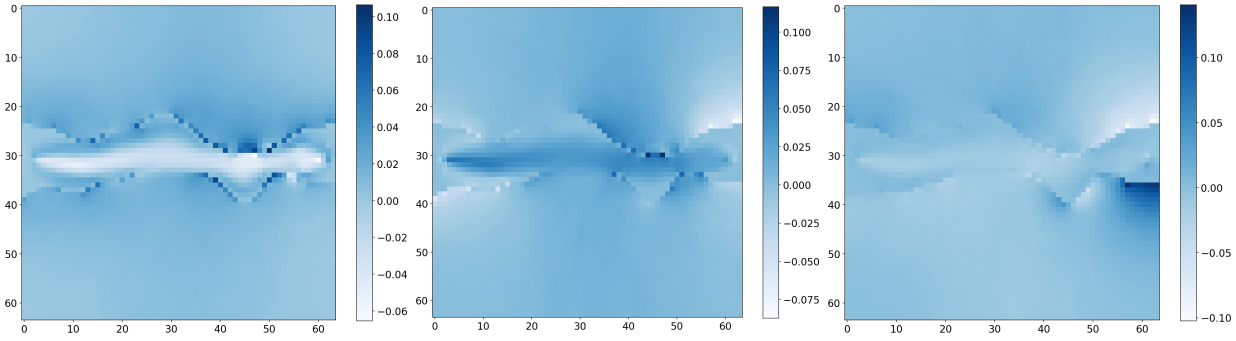
(a) First Eigenvector (b) Second Eigenvector (c) Third Eigenvector

Figure 6: First three largest eigenvector of FIM



(a) First Eigenvector (b) Second Eigenvector (c) Third Eigenvector

Figure 7: First three largest eigenvector of FIM



(a) First Eigenvector (b) Second Eigenvector (c) Third Eigenvector

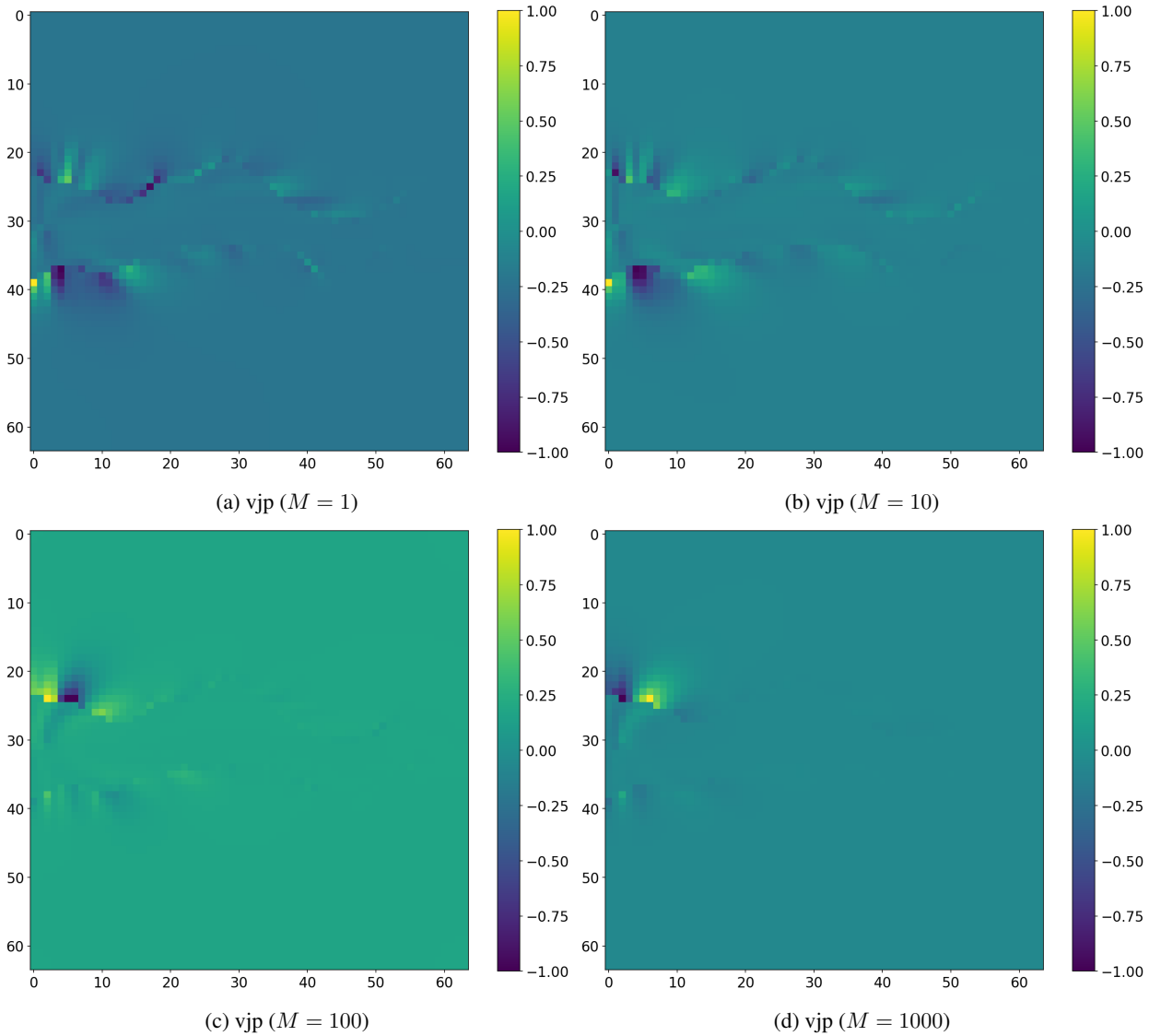
Figure 8: First three largest eigenvector of FIM

4.2.3.3 $M = 100$

4.2.3.4 $M = 1000$

- As M increases, we observe flow through the channel clearer.
- We see the boundary of permeability gets clearer.
- In general, it gets less noisy.

4.2.4 How does vector Jacobian product look like as M increases?



(a) vjp ($M = 1$)

(b) vjp ($M = 10$)

(c) vjp ($M = 100$)

(d) vjp ($M = 1000$)

Figure 9: Normalized Vector Jacobian Product when vector is the largest eigenvector

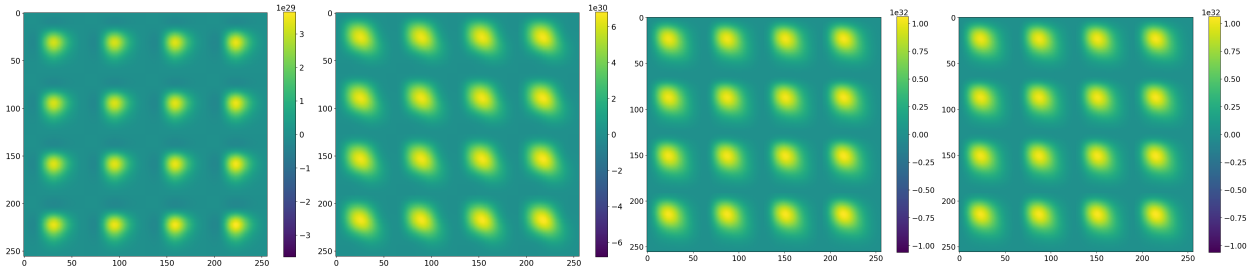
- We observe that vector Jacobian product looks more like saturation rather than permeability.
- As M increases, scale in color bar also increases.
- One possible conclusion:
  - vjp tells us the location in the spatial distribution (likelihood space) where there exists the largest variation, thus have the most information on parameter.
  - $J^T v$, when v is the largest eigenvector of FIM, is projecting Jacobian onto direction of maximum sensitivity.

4.3 When Random Variable of FIM, Y, is only Saturation

After updating the code, we compute FIM of saturation only.

4.3.1 FIM obtained

- We observe that we see off-diagonal structure in this Fisher Information Matrix.
- This just means that that are dependency or stronger correlation between parameters.
- This might be due to the structure of permeability being heterogenous, where point outside the channel does not impact saturation at all.



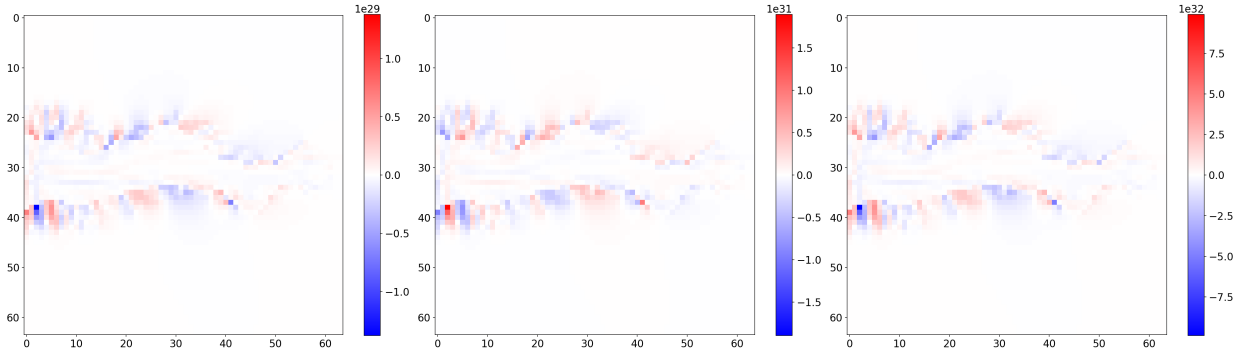
(a) $M = 1$ (b) $M = 10$ (c) $M = 100$ (d) $M = 1000$

Figure 10: FIM[:256, :256] of different M

4.3.2 The Each Rows of FIM

Each row of FIM can be considered as some linear combination of gradient. Each row represents each grid point of permeability that is perturbed, and the plot we are seeing shows how likelihood changes when the certain grid point of permeability is perturbed.

When $M = 1$,



(a) $i = 1$ (b) $i = 500$ (c) $i = 2000$

Figure 11: FIM of each rows when $M = 1$

When $M = 10$,

When $M = 100$,

When $M = 1000$,

4.3.3 Eigenvector of FIM

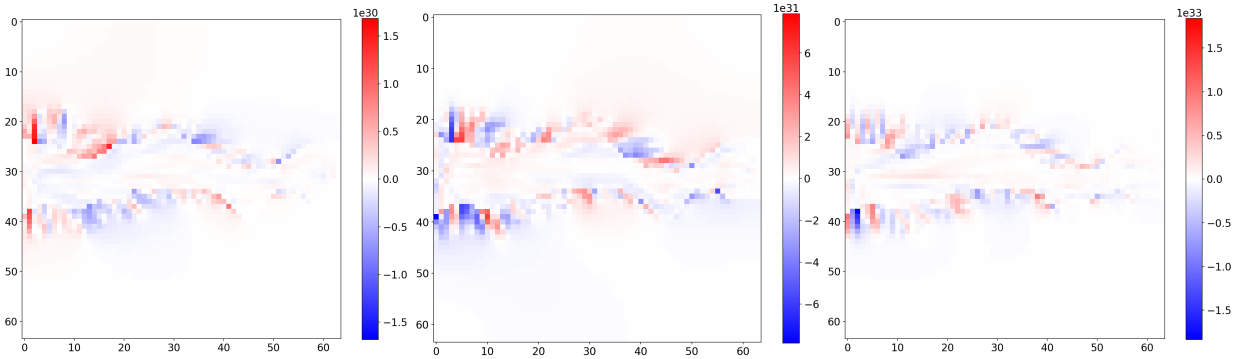
4.3.4 Vector Jacobian Product Obtained

4.3.5 Eigenvector of FIM

5 Training Result

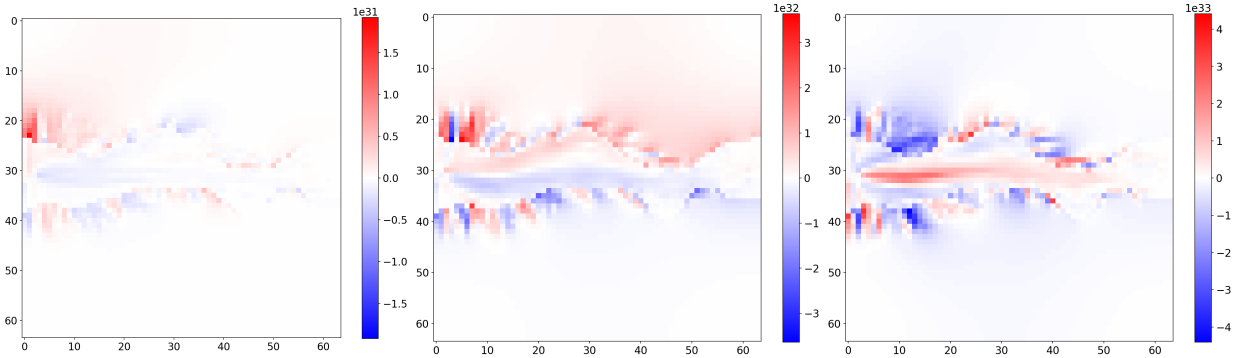
We first training with following configuration:

- Training , Test = [1800, 200]



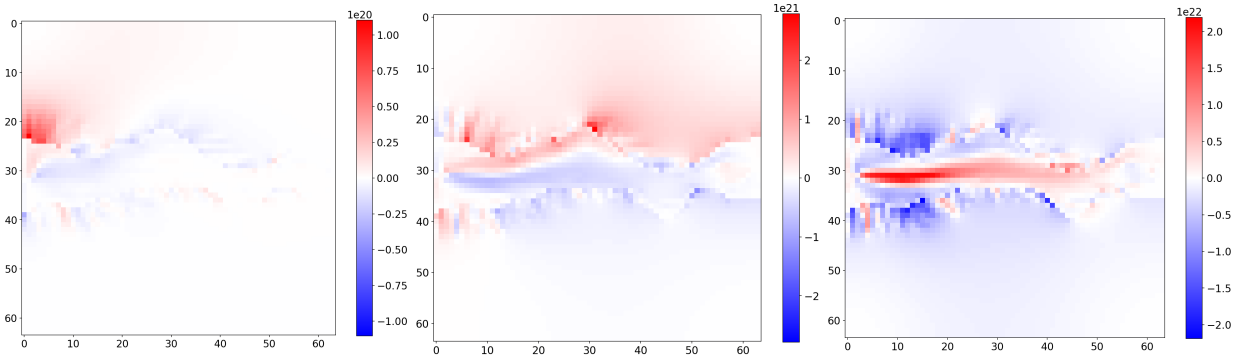
(a) $i = 1$ (b) $i = 500$ (c) $i = 2000$

Figure 12: FIM of each rows when $M = 10$



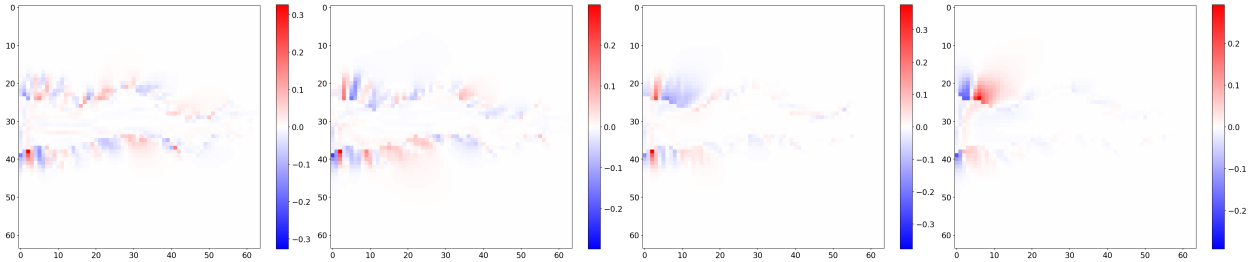
(a) $i = 1$ (b) $i = 500$ (c) $i = 2000$

Figure 13: FIM of each rows when $M = 100$



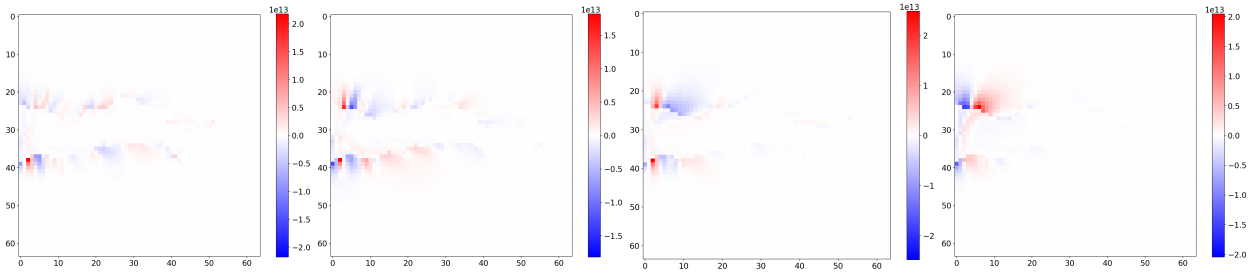
(a) $i = 1$ (b) $i = 500$ (c) $i = 2000$

Figure 14: FIM of each rows when $M = 1000$



(a) $M = 1$ (b) $M = 10$ (c) $M = 100$ (d) $M = 1000$

Figure 15: The largest eigenvector of FIM of different M



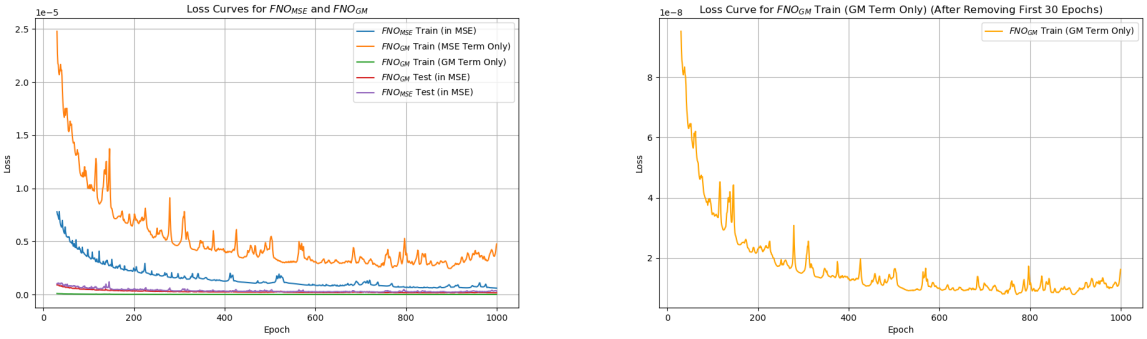
(a) $M = 1$ (b) $M = 10$ (c) $M = 100$ (d) $M = 1000$

Figure 16: The largest eigenvector of FIM of different M

- Batch size = 100
- Number of Epoch = 1000

Table 1: Loss Table

	Train Loss	Test Loss
	MSE/GM	MSE
FNO_{MSE}	3.3622×10^{-8}	8.4016×10^{-8}
FNO_{GM}	2.6428×10^{-7}	1.5976×10^{-7}



(a) All loss (b) Only GM Term

Figure 17: Loss plots

5.1 MSE

5.1.1 Forward Simulation

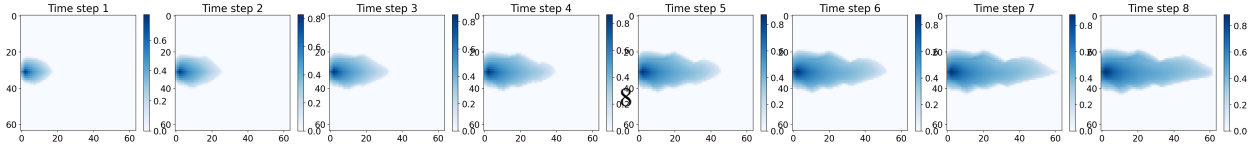


Figure 18: True Saturation

1. Scale in the color bar does not match.
2. The learned vjp looks noisy as there are some colors showing in the part where it should be just white.

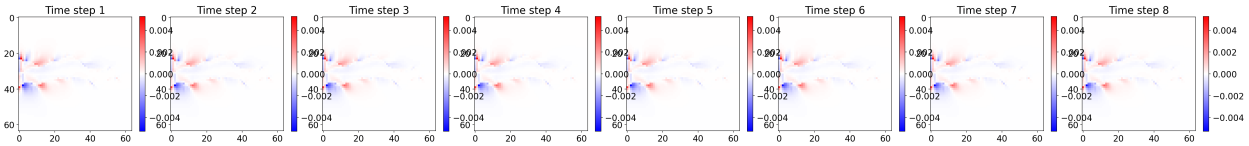


Figure 21: True vjp

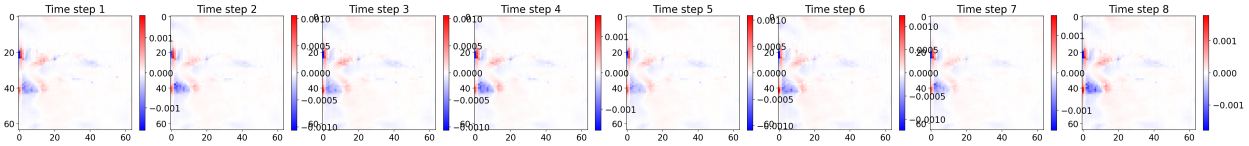


Figure 22: Learned vjp

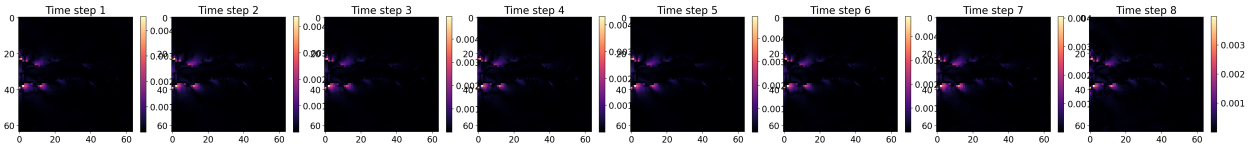


Figure 23: Absolute Difference

5.2 Gradient-Matching

5.2.1 Forward Simulation

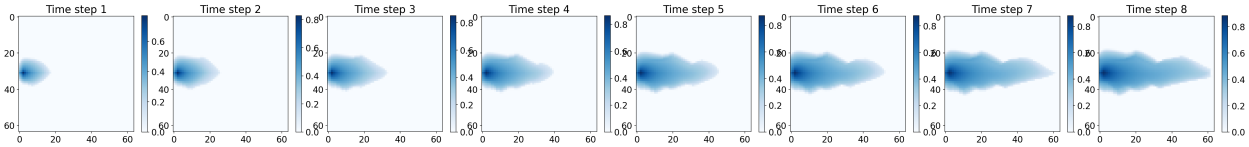


Figure 24: True Saturation

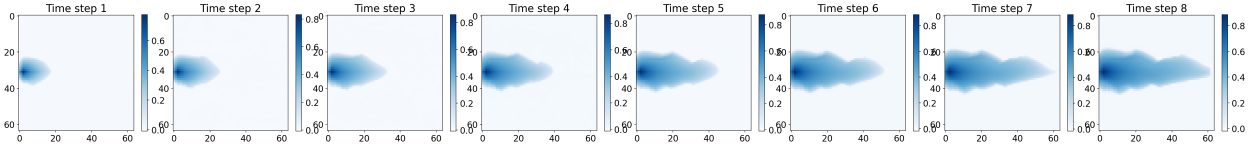


Figure 25: Predicted Saturation

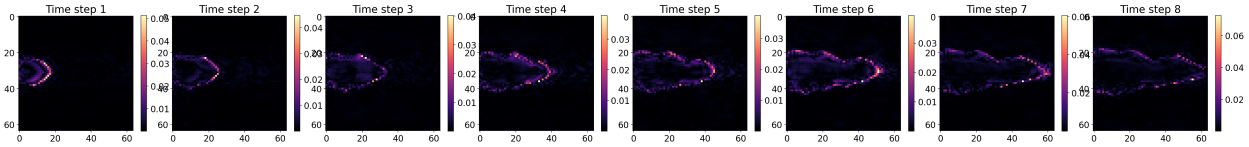


Figure 26: Absolute Difference

5.2.2 Learned and True vjp

We now observe that the learned and the true vjp matches well. Unlike MSE model, we observe

1. The scale of color bar matches correctly.
2. The plot does not look noisy.

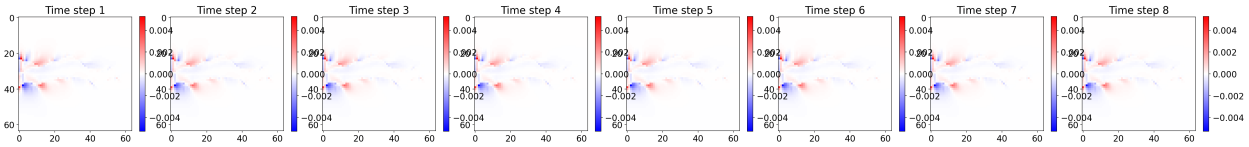


Figure 27: True vjp

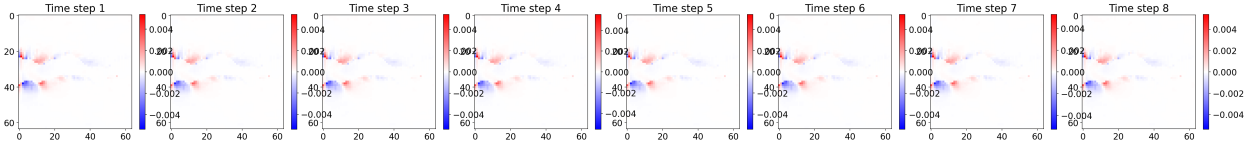


Figure 28: Learned vjp

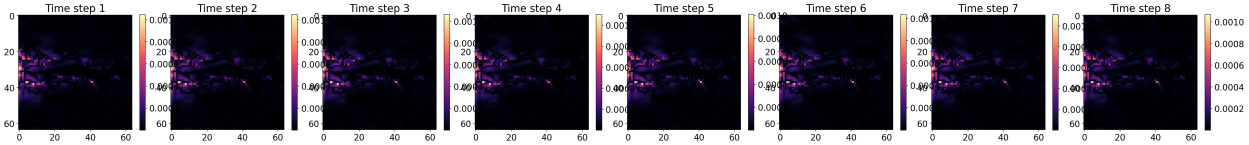


Figure 29: Absolute Difference

5.3 Future Step

1. TODO: Debug NS eigenvector and vjp.
2. TODO: Want to generate the full dataset for Francis' dataset (which might take 1 or 2 days).
3. TODO: Try it on Jason's dataset (Now that we fixed the problem with FIM computation, we are optimistic about the experiment, so we want to try it again.)

5.4 Question

1. Do we want to train both models for a longer time?