Comparison of ML-Based Proxy Modeling Strategies

Lessons Learned from the SMART Initiative

Jared Schuetter
Data Scientist
Battelle Memorial Institute

U.S. DEPARTMENT OF ENERGY

NATIONAL ENERGY TECHNOLOGY LABORATORY

Disclaimer

NATIONAL ENERGY TECHNOLOGY LABORATORY

This project was funded by the United States Department of Energy, National Energy Technology Laboratory, in part, through a site support contract. Neither the United States Government nor any agency thereof, nor any of their employees, nor the support contractor, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Authors and Contact Information

Jared Schuetter[1,3]; Chung Shih[1]; Paul Holcomb[1,2]; Hongkyu Yoon[4]; Meen Kadeethum[4]; Alexandre Tartakovsky[5]; Christian Muñoz Oro[5]; Seyyed Hosseini[6]; Hongsheng Wang[6]

[1]National Energy Technology Laboratory, 626 Cochran Mill Road, Pittsburgh, PA 15236, USA

[2]NETL Support Contractor, 626 Cochran Mill Road, Pittsburgh, PA 15236, USA

[3]Battelle Memorial Institute, 505 King Ave, Columbus, OH 43201, USA

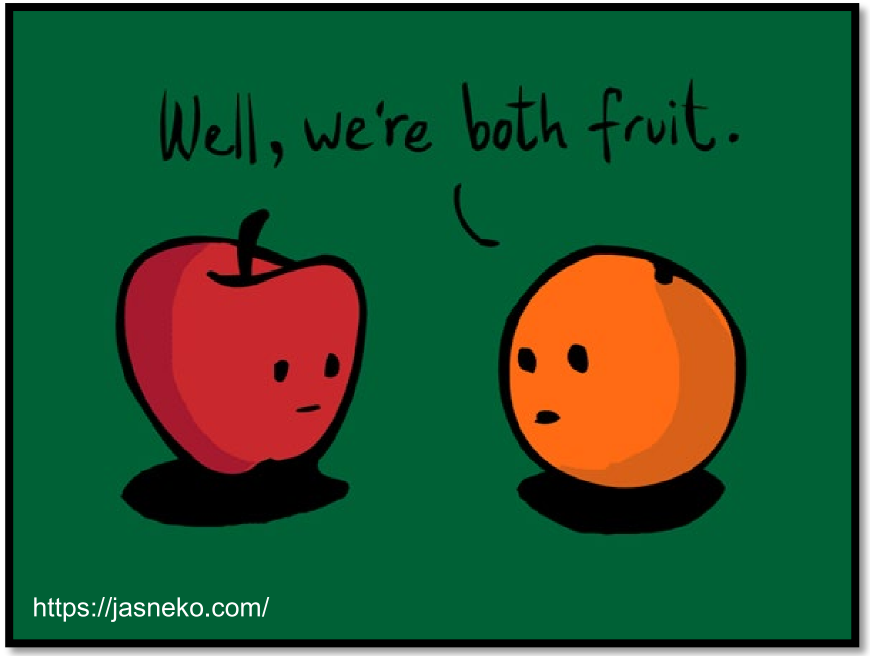
[4]Sandia National Laboratories, 1515 Eubank SE, Albuquerque, NM 87123, USA

[5]University of Illinois Urbana-Champaign, 205 North Mathews Avenue, Urbana, IL 61801, USA

[6]Univ. of Texas – Bureau of Economic Geology, 10100 Burnet Road., Building 130, Austin, TX 78758, USA

Background and Objectives

- For some research problems, SMART has generated multiple candidate solutions
- These solutions typically use different code bases, are developed by different organizations, and in some cases are built from different training and validation datasets
- To evaluate the strengths and weaknesses of the solutions, it is necessary to compare their performance on the same datasets using the same metrics
- The goal of the subtask described in this presentation was to compare several machine learning (ML) based reservoir proxy models for the Illinois Basin - Decatur Project (IBDP)
- This talk will describe what was done and share lessons learned from that activity



https://jasneko.com/

General Considerations Comparing Models

- Our goal was to make the analysis understandable, repeatable, and flexible
- Common strategies (see below) were used to try to ensure these goals were met

Understandable

- Two-Way Communication Between Teams
- Use Only The Data That Are Needed
- Use Multiple Metrics

Understandable / Repeatable & Flexible

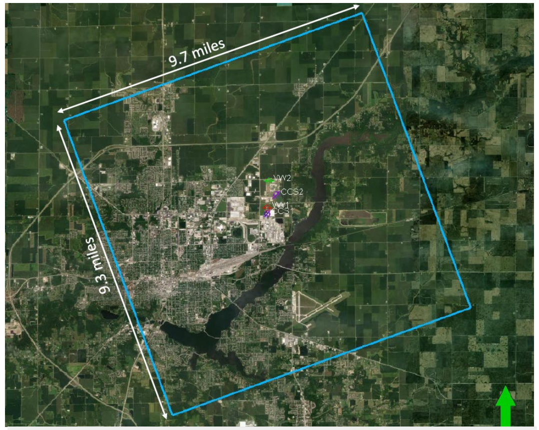
- Use A Common Comparison Script
- Use Modular Code

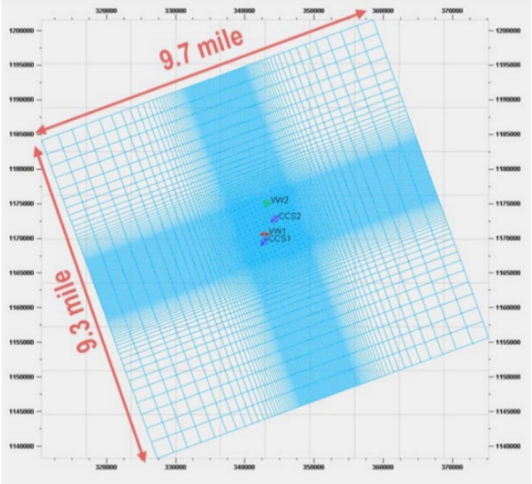
Repeatable & Flexible

- Create APIs For Model Predictions

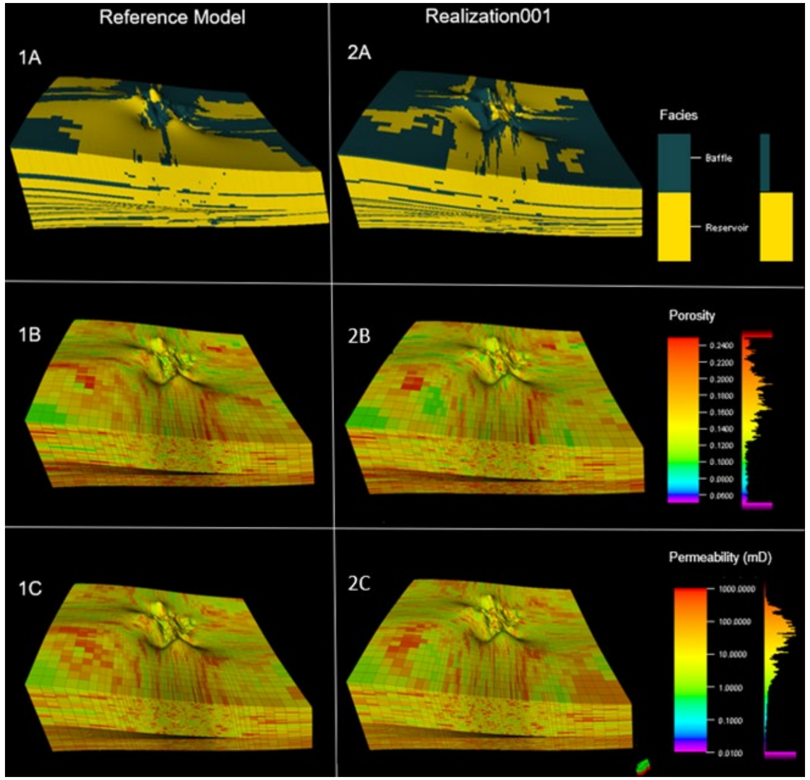
IBDP Simulation Model

- The Comparison: Compare four forward models built on the same train and test sets





IBDP Site and Reservoir Model Details	
Characteristic	**Value**
Model Software	Eclipse
Geologic Inputs	Porosity/perm realizations from a spatial model
Reservoir Size and Shape	Tartan Grid (126x, 125y, 110z) z = 1 is the surface
Timepoints	50 (monthly)
Injection Well Location	(x = 54, y = 76)
Injection Well Packer-Separated Perforation Zones	Upper: z = 74 Middle: z = 76-81 Lower: z = 83-86
Monitoring Well Location	(x = 57, y = 68)
Monitoring Well Sensor Depths	z = 29, 43, 62, 79, 84, 91



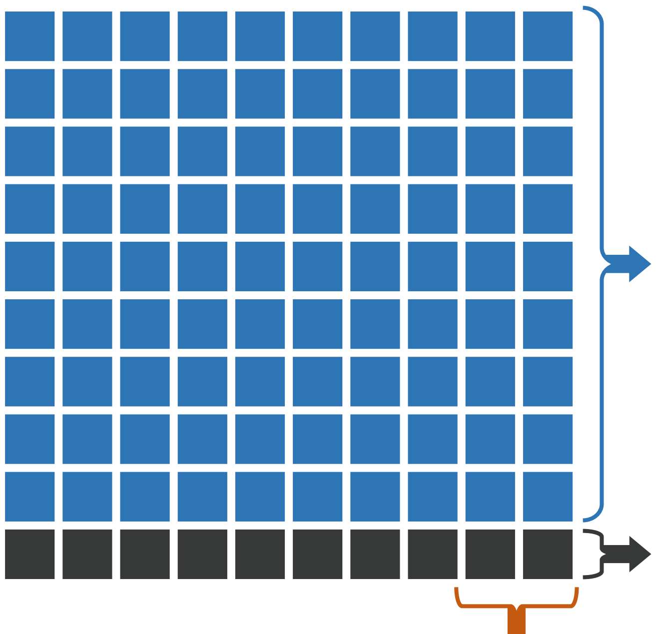
Example IBDP realization (right column) compared to the IBDP reference model (left column).

Simulation Data for Proxy Model Training & Evaluation

NATIONAL ENERGY TECHNOLOGY LABORATORY

- The Comparison: Compare four forward models built on the same train and test sets

100 IBDP Eclipse simulation runs, differing only in terms of the geology (porosity/permeability distribution, baffle locations, and transmissivity modifiers)



90 runs used to train each of the models

10 runs used to evaluate each of the models

20 runs used different transmissivity modifiers.

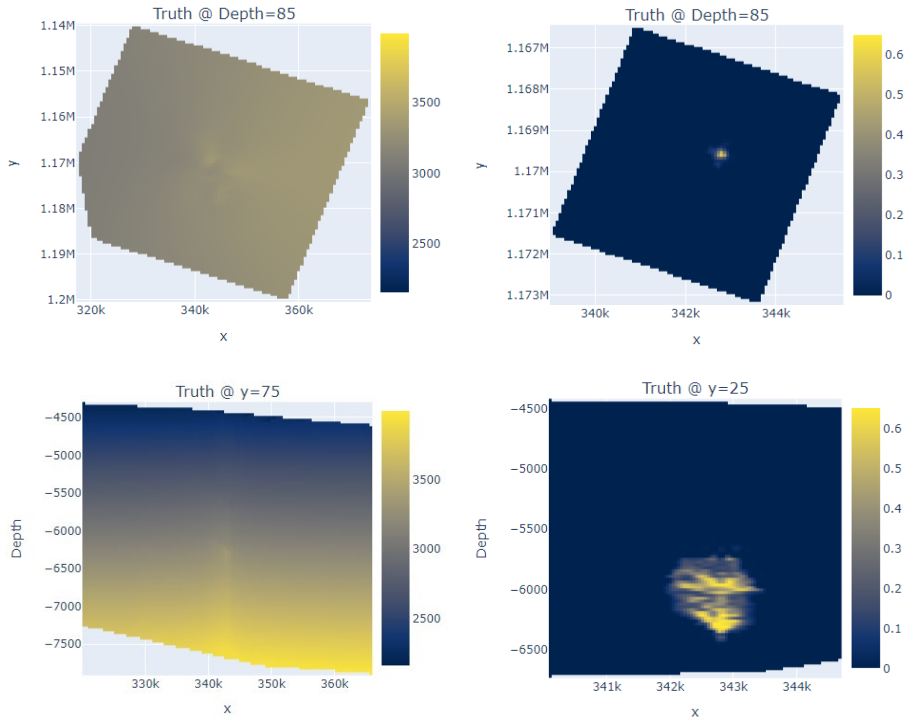
Simulation Output = Spatiotemporal Pressure & Saturation

Pressure

Saturation

Horizontal Slice at the Lower Injection Zone (t = 30)

Vertical Slice Through the Injection Well (t = 30)



Proxy Models and What is Needed To Compare Them

- The Comparison: Compare four forward models built on the same train and test sets

Candidate Models	

Natl. Energy Technology Laboratory
• Long short-term memory (LSTM)
• Pressure model:
  ▪ Fully connected MLP layers
• Saturation model:
  ▪ Fully connected MLP layers
• Both models predict on cropped domain [32:96, 32:96, 29:84]
	Sandia Natl. Laboratory
	• Improved neural operator (iNO)
	• Encoder/decoder used to build model in a latent space
	• Predictions can be made at any time or location within domain

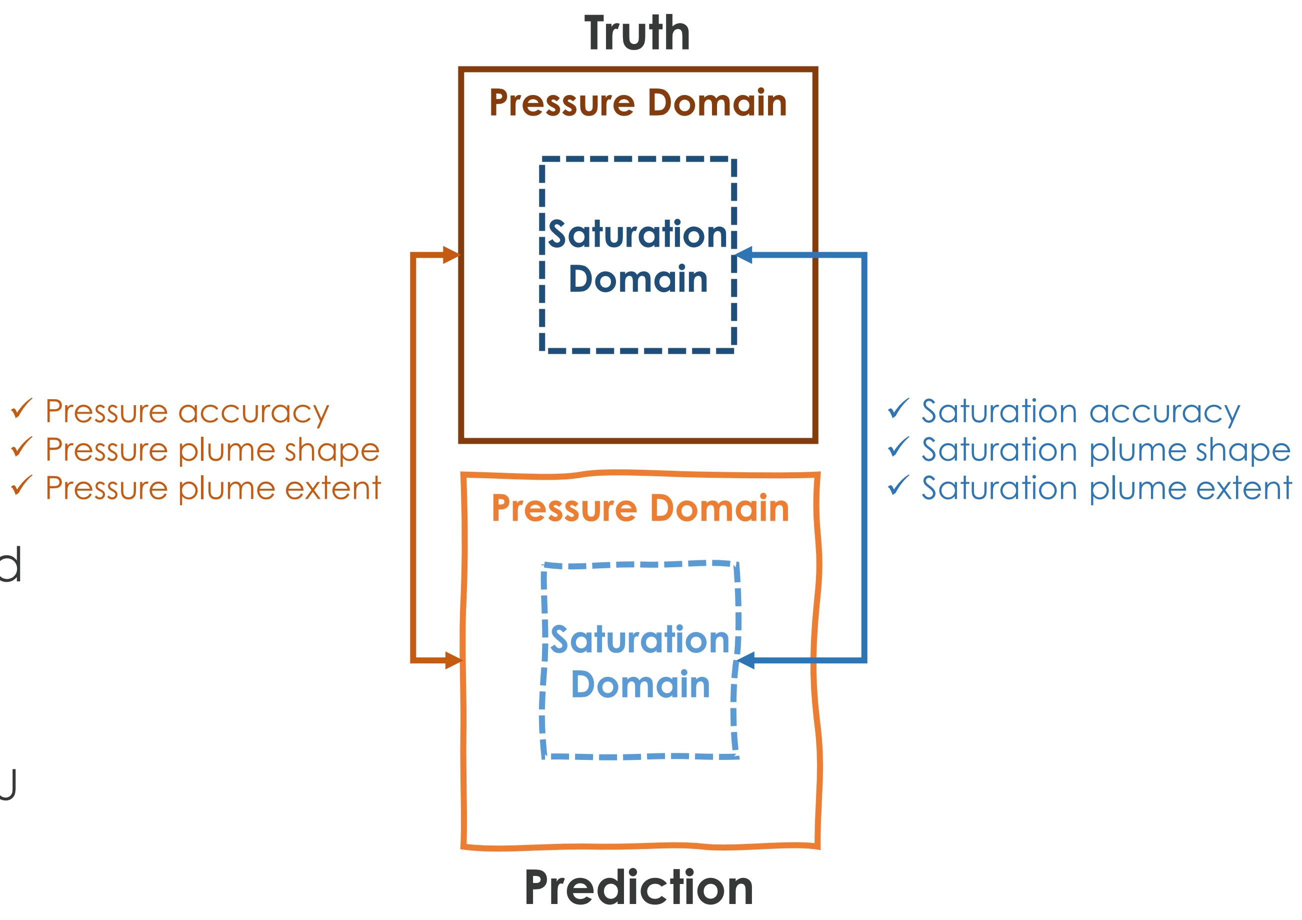
U Illinois Urbana-Champaign
• Karhunen-Loeve Deep Neural Network (KL-DNN)
• Dimension reduction through KL expansion, modeling in that space
• Pressure predictions: Full domain
• Saturation predictions: Cropped domain [31:70, 51:94, 1:94]
	U Texas - Bureau of Econ. Geology
	• U-Net
	• Pressure model:
	  ▪ Fully connected MLP layers
	  ▪ Prediction on full domain
	• Saturation model:
	  ▪ Convolutional layers
	  ▪ Prediction on cropped domain [27:75, 49:97, 1:97]

- Desired Comparisons:
  - Pressure prediction accuracy across the model domain
  - Saturation prediction accuracy across the active subset of the reservoir
    [x = 31:70, y = 51:94, z = 1:94]
  - Agreement in pressure and saturation plume shape and magnitudes
  - Agreement in pressure and saturation plume extent, especially as it relates to Area of Review calculation
  - Training and inference speed, computational burden, hardware requirements, etc.

How We Used the Comparison Strategies

- Comparison Strategy: **Use Only The Data That Are Needed**
  - "Truth Data" for each of 10 test cases
    - **IBDP simulated pressure** over full domain
    - **IBDP simulated saturation** over sub-domain
  - "Prediction Data" for each model and test case
    - **Predicted pressure** over full domain
    - **Predicted saturation** over sub-domain
  - "Computational Data" for each model and sub-model (pressure, saturation)
    - **Training hardware** and average CPU/GPU **training time** using 90 training cases
    - **Inference hardware** and average CPU/GPU **inference time** over the 10 test cases

✓ Training/Inference Speed
✓ Computational burden
✓ Hardware requirements



How We Used the Comparison Strategies

- Comparison Strategies:
  Create APIs For Model Predictions, Two-Way Communication Between Teams
  - Worked with the modeling teams to create an API for prediction data
  - Data were provided in HDF-5 file format (.h5) with the nested structure shown below

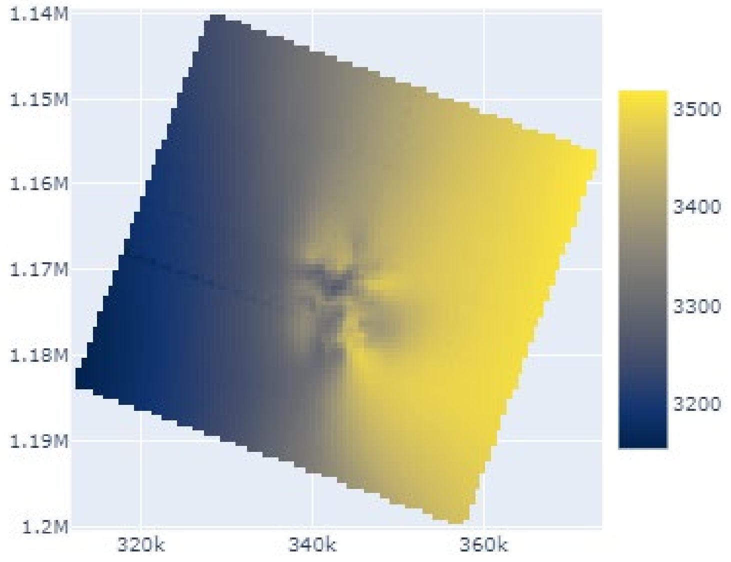
Example code to generate a 2D horizontal slice (`pressure_slice_12`) of the predicted pressure volume at depth = 93 and time = Month 12 for Run 10.

```
import h5py
f = h5py.File(path_to_h5_file, 'r')
pressure_data = f['Run 10']['Pressure']
pressure_slice_12 = pressure_data[,,92,11]
```

Case Number — String of the form "Run XX", where XX is the run number

Output Type — "Permeability", "Porosity", "Pressure", "Saturation", "Production" (if applicable)

Data Array — NumPy array of the form [x, y, z, t], where (x, y, z) is the cell location in x, y, and depth (z), and t is the timepoint



How We Used the Comparison Strategies

- Comparison Strategy: **Use Multiple Metrics**
  - These were all regression models producing outputs at each cell in the reservoir
  - Standard metrics are root mean squared error (RMSE) and mean absolute error (MAE)
  - In this case, we also want to be able to understand how residuals change across the volume, so we used weighted versions of these metrics:

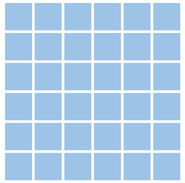
Weighted RMSE

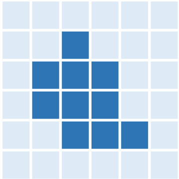
$$RMSE = \sqrt{\frac{\sum_{i \in S} w_i^2 (y_i - \hat{y}_i)^2}{|S|}}$$

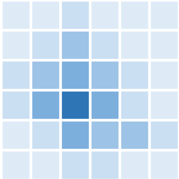
Weighted MAE

$$MAE = \frac{\sum_{i \in S} w_i |y_i - \hat{y}_i|}{|S|}$$

Weighting Scheme	Weights w_i	Set S
Classical (uniform) weighting	$w_i = 1 \ \forall i \in S$	$S = \{all\ cells\}$
Non-zero (NZ) weighting	$w_i = 1 \ \forall i \in S$	$S = \{i: y_i \neq 0\}$
Rate of Change (RoC) weighting	$w_i = \|S\| \cdot \frac{\|y_{i,t+1} - y_{i,t}\|}{\sum_{j \in S} \|y_{j,t+1} - y_{j,t}\|}$ $y_{j,t}$ is the value in cell j at time t	$S = \{i: y_i \neq 0\}$







How We Used the Comparison Strategies

NATIONAL ENERGY TECHNOLOGY LABORATORY

- Comparison Strategies: **Use a Common Comparison Script, Use Modular Code**
  - The script works from a dictionary of files
  - After specifying their names, they could all be loaded and analyzed the same way because of the standard data format that we agreed upon

```
# Specify Data Locations
truthFile = dataDir + 'Truth.h5'
reservoirFile = dataDir + 'ibdp.h5'
modelResultFiles = {'UTBEG': dataDir + 'ut_beg_data.h5',
                    'SNL': dataDir + 'snl_results.h5',
                    'UIUC': dataDir + 'uiuc_data_mean.hdf5',
                    'NETL': dataDir + 'netl_ibdp_lstm.h5'}
resultFile = projDir + 'Comparison_Results.h5'
```

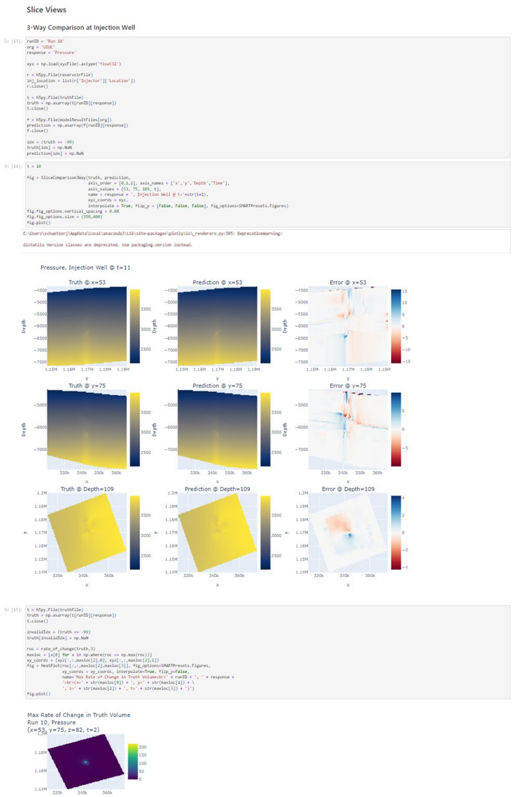
New models can be incorporated by making a single change to the *modelResultFiles* dictionary

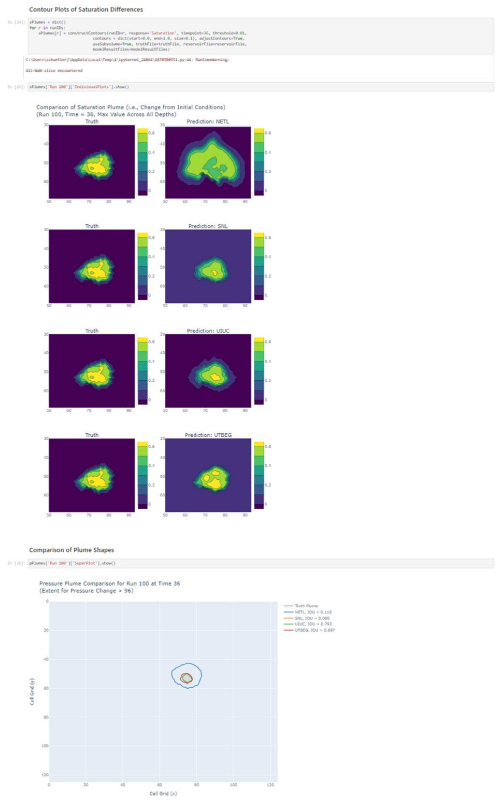
```
# Load Datasets
reservoir = h5py.File(reservoirFile, 'r')
truth = h5py.File(truthFile, 'r')
prediction = dict()
for k in modelResultFiles.keys():
    prediction[k] = h5py.File(modelResultFiles[k], 'r')
```

"truth" and "prediction" go into a loop through the models where comparisons are made using a *SMARTComparer* class on appropriate sub-volumes of the reservoir

How We Used the Comparison Strategies

- Comparison Strategy: **Use a Common Comparison Script**
  - Scripts are in a Jupyter notebook to more easily re-run the comparison in the future
  - Text could be embedded here as well, if needed, to produce an interactive report





Comparison Results – Macro Level

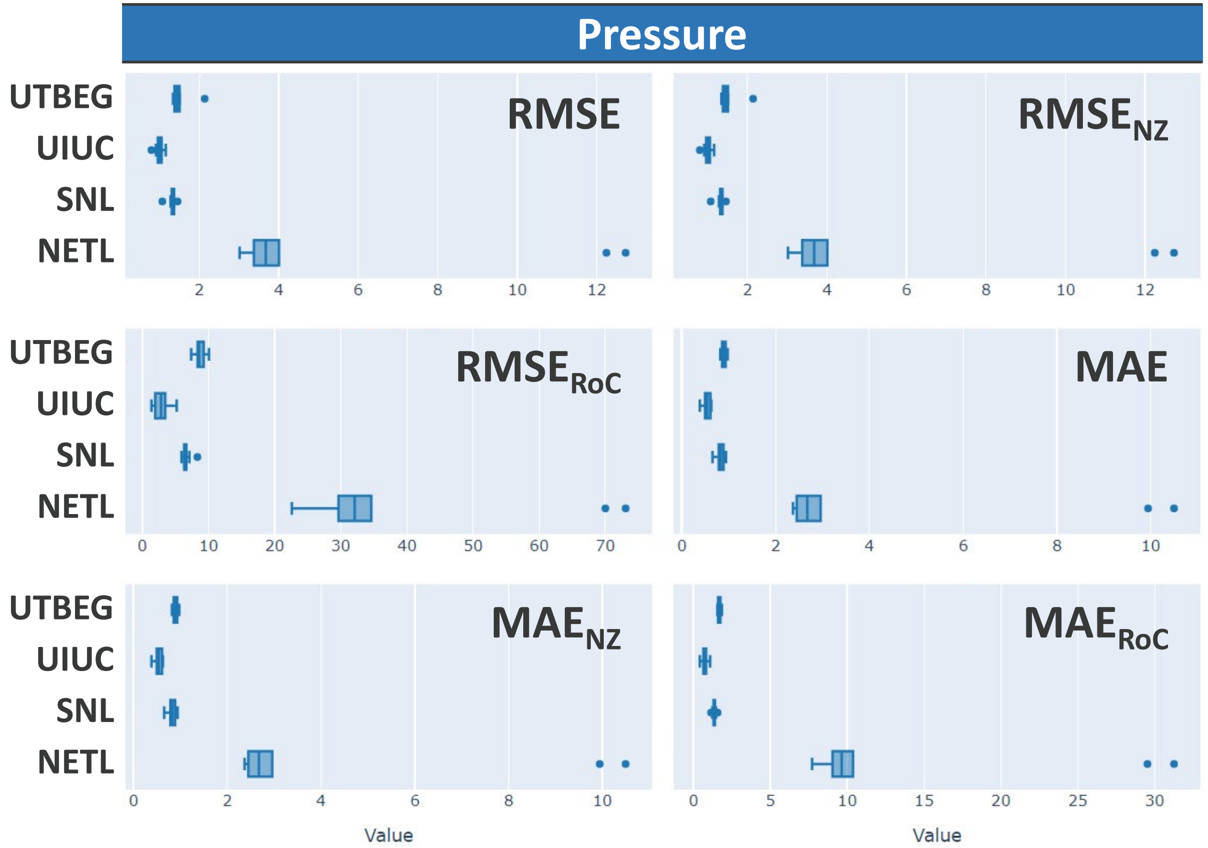
- Overall Model Accuracy
  - Global metrics calculated across all 10 test runs, 50 timesteps, and 1.7M grid cells
  - UIUC's KL-DNN is the top performer across the board for pressure prediction
  - UTBEG's U-Net is the best model for saturation prediction

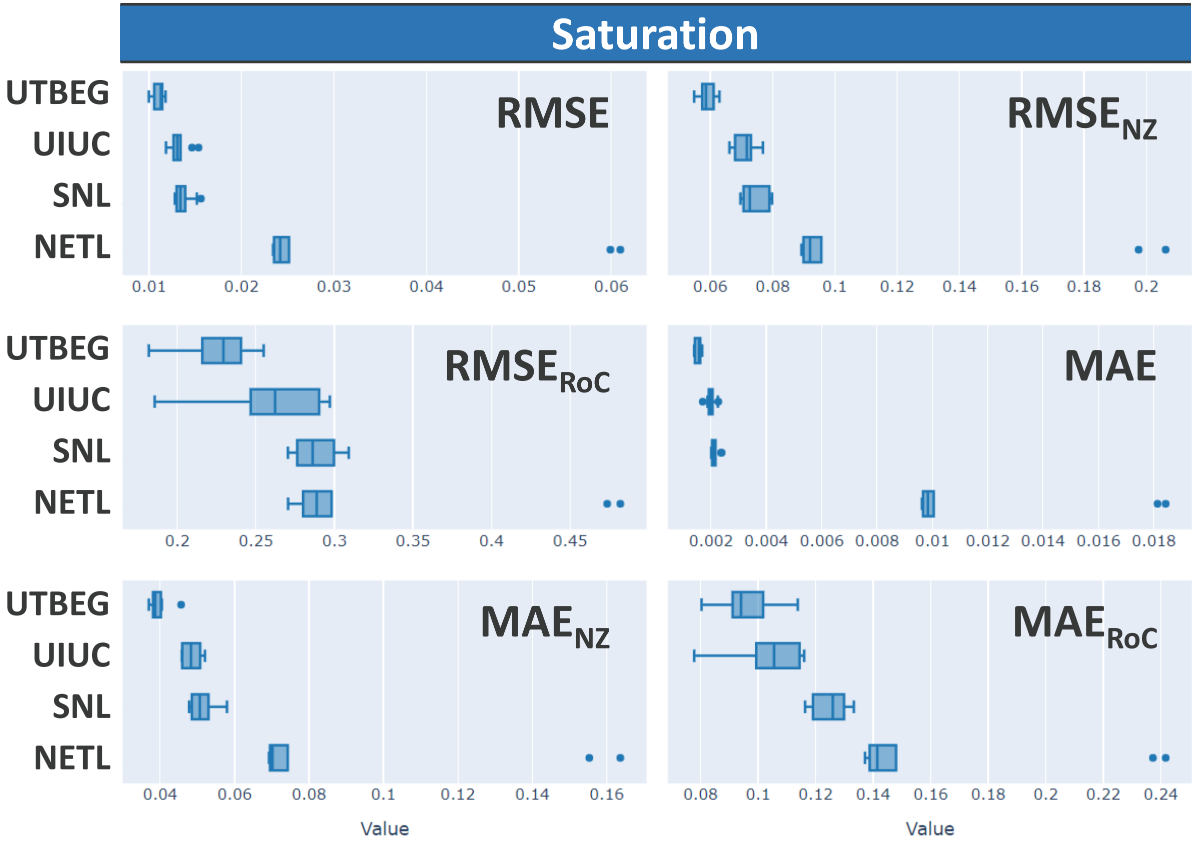
Response	Model	RMSE	RMSE_NZ	RMSE_RoC	MAE	MAE_NZ	MAE_RoC
Pressure	NETL	5.602	5.602	76.228	4.133	4.133	13.475
	SNL	1.360	1.360	12.619	0.822	0.822	1.351
	UIUC	***1.022***	***1.022***	***5.019***	***0.537***	***0.537***	***0.743***
	UTBEG	1.560	1.560	19.803	0.895	0.895	1.675
Saturation	NETL	0.033	0.122	0.267	0.011	0.094	0.101
	SNL	0.015	0.079	0.222	***0.002***	0.053	0.072
	UIUC	0.015	0.077	0.220	***0.002***	0.052	0.070
	UTBEG	***0.012***	***0.064***	***0.183***	***0.002***	***0.041***	***0.056***

- Note: These results can be misleading since most of the reservoir has zero saturation and small pressure change for most timesteps… RoC weighting was meant to account for this.

Comparison Results – By Run

- Pressure and Saturation Accuracy by Run
  - Generally consistent performance with more inter-model variability than intra-model variability
  - Larger errors associated with Runs 90 and 100, which had different transmissivity modifiers



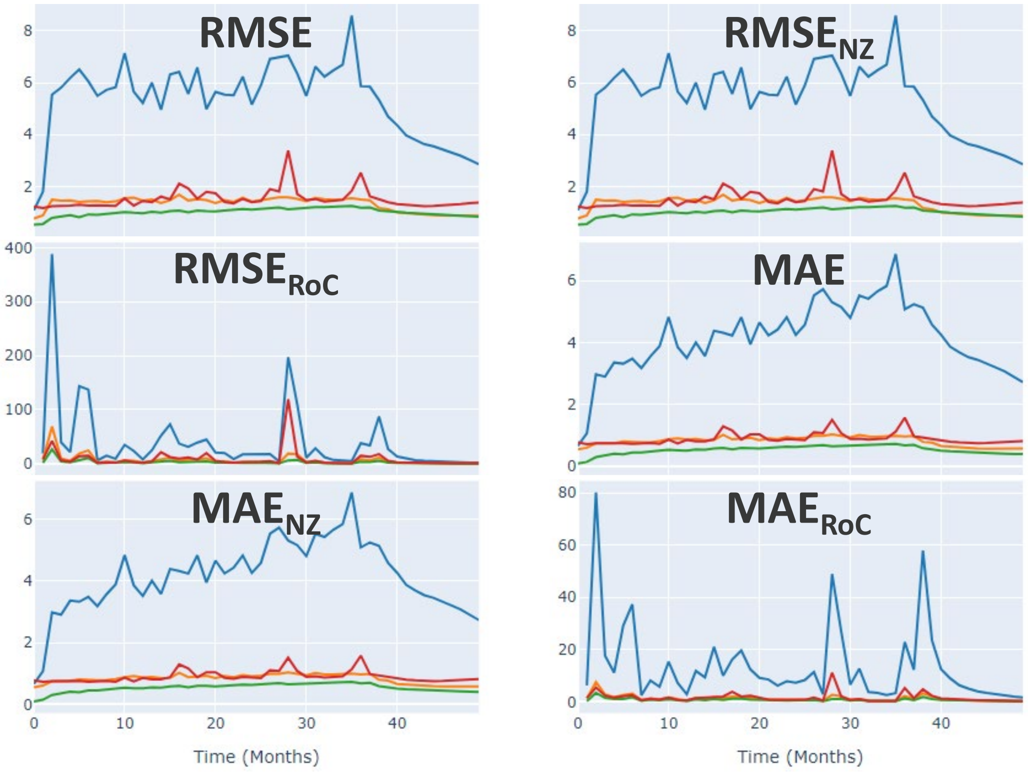


Comparison Results – Pressure Over Time

NATIONAL ENERGY TECHNOLOGY LABORATORY

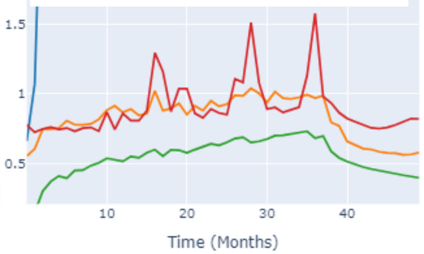
- Pressure and Saturation Accuracy by Timestep
  - Errors increase through the injection period, then tail off around the end of injection (month 36)
  - Spikes in error around months 16, 28, 36, accentuated by the rate of change (RoC) metrics

Pressure

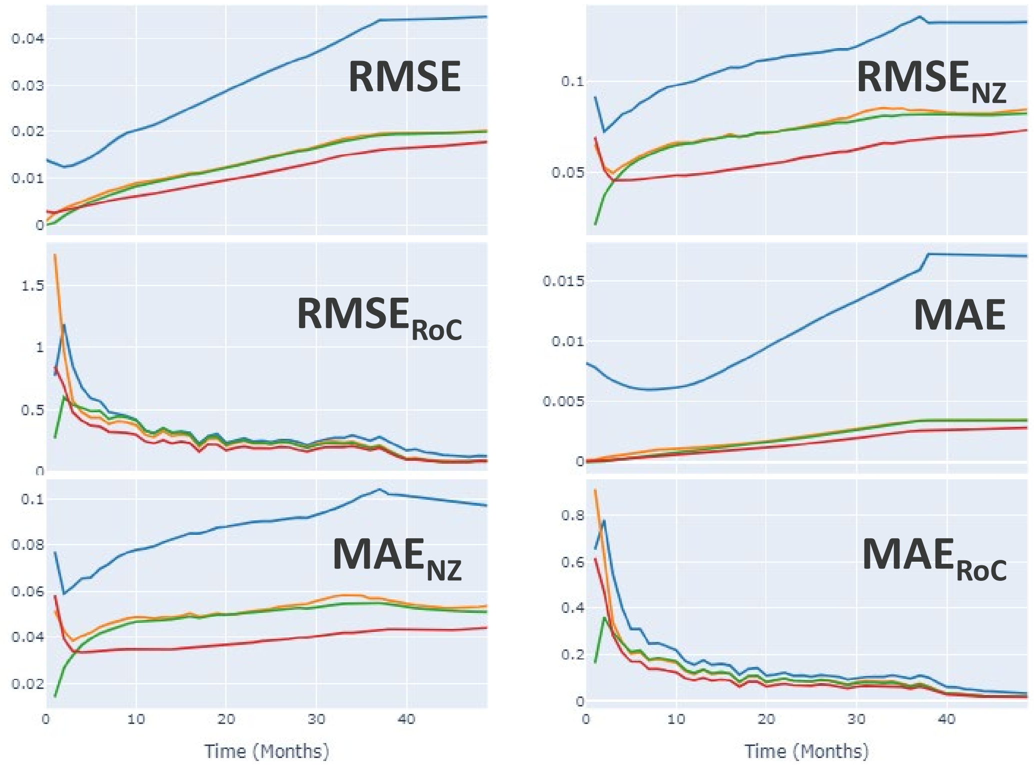


NETL
SNL
UIUC
UTBEG

Pressure MAE (Zoomed)

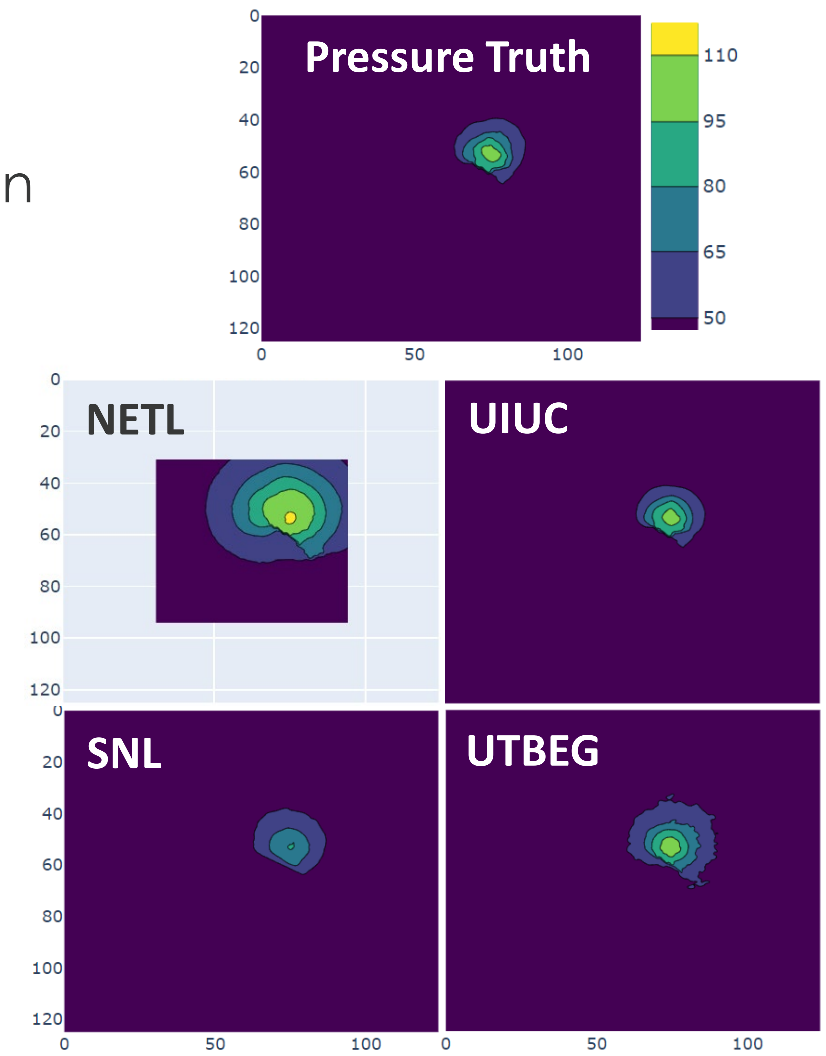


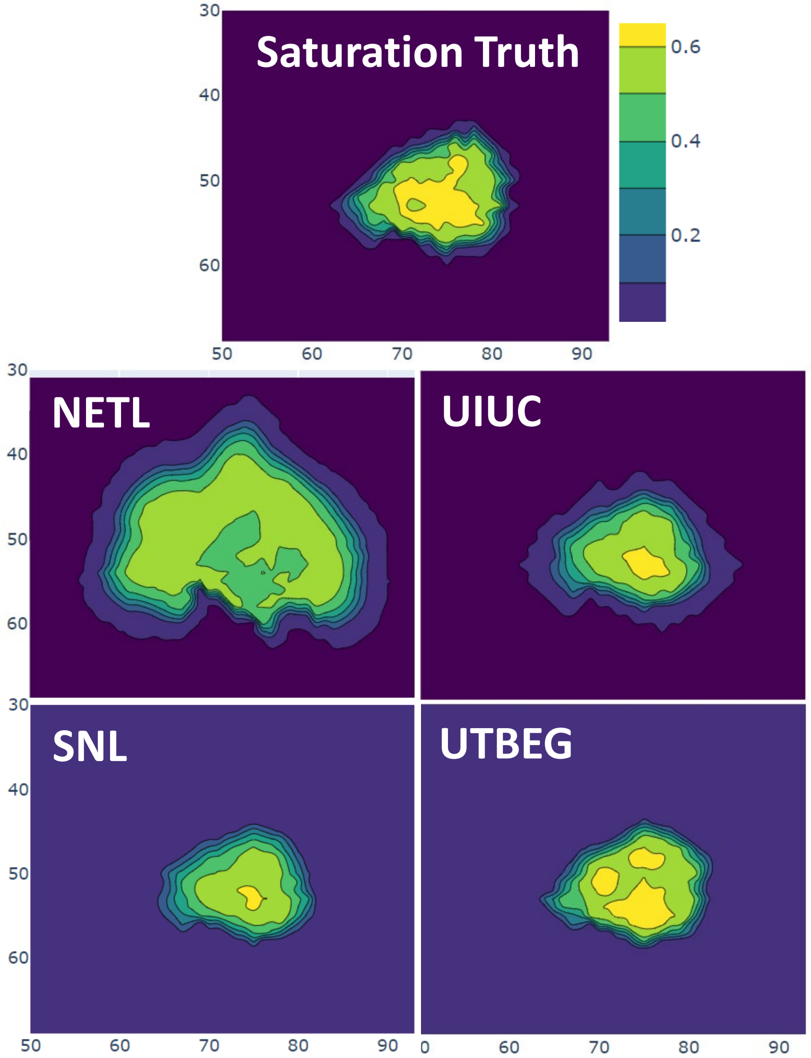
Saturation



Comparison Results – Plume Shape

- Contour plots were used to visualize the pressure and saturation plumes at end of injection (average value across z-dim)
- The best plumes come from the same overall best models
  - Pressure: UIUC
  - Saturation: UTBEG
- Note that saturation was only compared on the sub-volume of mostly active cells



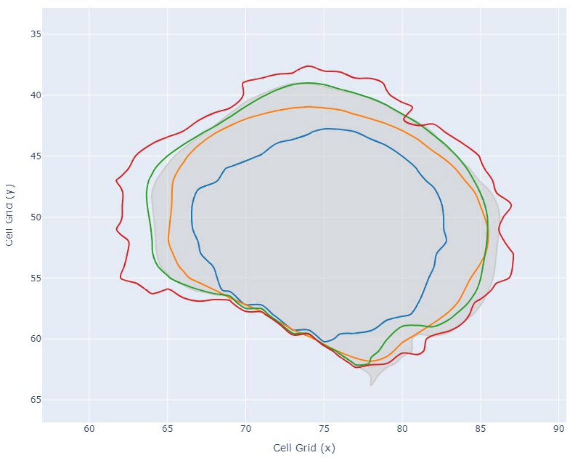


Comparison Results – Plume Extent

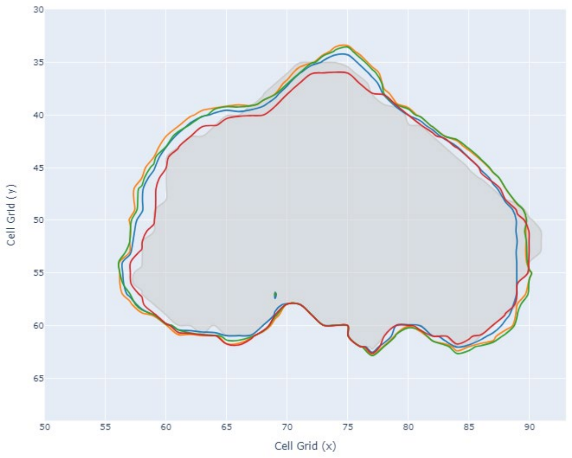
- Plume extent was defined by a critical threshold
  - Pressure ≥ 96 psi change
  - Saturation ≥ 0.01 (1%)
- Intersection-over-union (IOU) metric was used to measure agreement with ground truth

Truth
NETL
SNL
UIUC
UTBEG

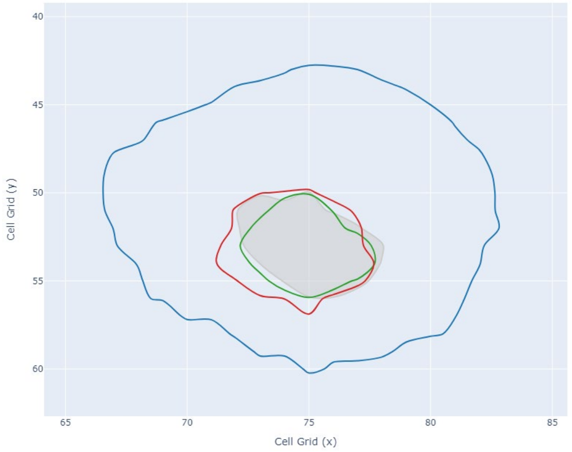
Pressure, Run 10



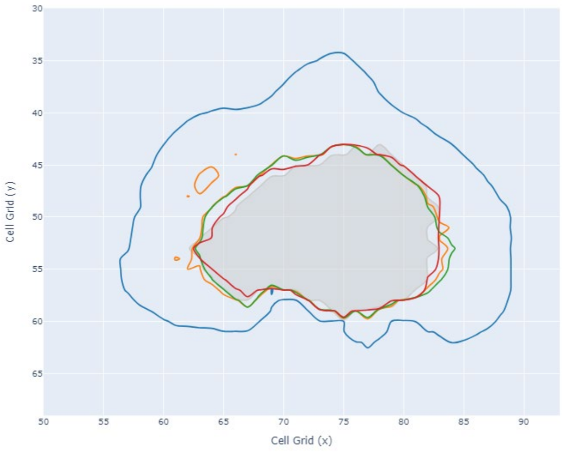
Saturation, Run 10



Pressure, Run 100



Saturation, Run 100



Pressure IOU

	NETL	SNL	UIUC	UTBEG
Run 10	0.570	0.832	**0.924**	0.853
Run 20	0.591	0.744	**0.947**	0.814
Run 30	0.666	0.855	**0.891**	0.713
Run 40	0.607	0.727	**0.938**	0.786
Run 50	0.558	0.784	**0.860**	0.852
Run 60	0.566	0.785	**0.917**	0.843
Run 70	0.676	0.836	**0.877**	0.713
Run 80	0.543	0.711	**0.881**	0.851
Run 90	0.193	0.000	0.475	**0.700**
Run 100	0.116	0.000	**0.792**	0.697

Saturation IOU

	NETL	SNL	UIUC	UTBEG
Run 10	0.882	0.855	0.855	**0.923**
Run 20	0.898	0.892	0.902	**0.928**
Run 30	0.901	0.865	0.860	**0.915**
Run 40	0.889	0.833	0.836	**0.908**
Run 50	0.798	0.768	0.778	**0.922**
Run 60	0.918	0.881	0.886	**0.923**
Run 70	0.860	0.820	0.820	**0.907**
Run 80	0.880	0.871	0.879	**0.888**
Run 90	0.347	0.815	**0.818**	0.800
Run 100	0.333	0.817	0.833	**0.897**

Comparison Results – Computational Burden

- Teams provided run times and hardware used, but configurations were quite different
- Opted to use floating point operations per second (FLOPS) to convert all run times to the same hardware

Model		NETL	SNL	UIUC	UT-BEG
Training - Pressure	CPU/GPU Time	Not provided	149 min (1500 epochs)	~5 hours	~25 hrs
	Hardware	CPU	1x Quadro RTX 8000	1x NVIDIA RTX A5000	2x NVIDIA GeForce RTX 3090
Training - Saturation	CPU/GPU Time	Not provided	47 min (1000 epochs)	~5 hours	~14 hrs
	Hardware	CPU	1x Quadro RTX 6000	1x NVIDIA RTX A5000	2x NVIDIA GeForce RTX 3090
Inference - Pressure	CPU/GPU Time	2.403s (10 cases with 50 steps)	46.22s (including data transfer), 1s model eval	~2 seconds for all test cases	12.593s per realization
	Hardware	1x NVIDIA P100	1x Quadro RTX 8000	1x NVIDIA RTX A5000	2x NVIDIA GeForce RTX 3090
Inference - Saturation	CPU/GPU Time	1.989s (10 cases with 50 steps)	1.5s (incl. data transfer), 0.653s model eval	~2 seconds for all test cases	1.533s per realization
	Hardware	1x NVIDIA P100	1x Quadro RTX 6000	1x NVIDIA RTX A5000	2x NVIDIA GeForce RTX 3090

Hardware	FLOPS (FP32*)
NVIDIA P100	9.3
Quadro RTX 6000	16.3
Quadro RTX 8000	16.3
NVIDIA RTX A5000	27.8
GeForce RTX 3090	35.6
NVIDIA H100 SXM	67

* Single-precision floating point

Mode	Response	NETL	SNL	UIUC	UT-BEG
Training Time (Minutes)	Pressure	---	36.249	124.478	1594.030
	Saturation	---	11.434	124.478	892.657
Inference Time (Seconds)	Pressure	0.033	0.243	0.083	13.382
	Saturation	0.028	0.159	0.083	1.629

Concluding Remarks & Lessons Learned

- This comparison activity was mostly painless because:
  - There was planning and communication about how data would be delivered
  - The models were uniform (i.e., same training set, conditions, and output grid… mostly)
  - Ground truth was simulated, so we avoided a lot of complication found in real site characterization or field operation datasets (e.g., missing, inaccurate, or inconsistent data)
- For comparison tasks like this, it is crucial to consider how different approaches will be compared <u>before</u> planning the task where they are implemented
- Thanks to all the modeling teams for fulfilling my many requests for information and working hard to provide results in the formats needed to do the comparisons!
  - NETL: Chung Shih, Paul Holcomb
  - SNL: Hongkyu Yoon, Meen Kadeethum
  - UIUC: Alex Tartakovsky, Christian Munoz Oro
  - UTBEG: Seyyed Hosseini, Hongsheng Wang

NETL RESOURCES

VISIT US AT: **www.NETL.DOE.gov**

@NETL_DOE

@NETL_DOE

@NationalEnergyTechnologyLaboratory

CONTACT:
Jared Schuetter
schuetterj@battelle.org

U.S. DEPARTMENT OF ENERGY