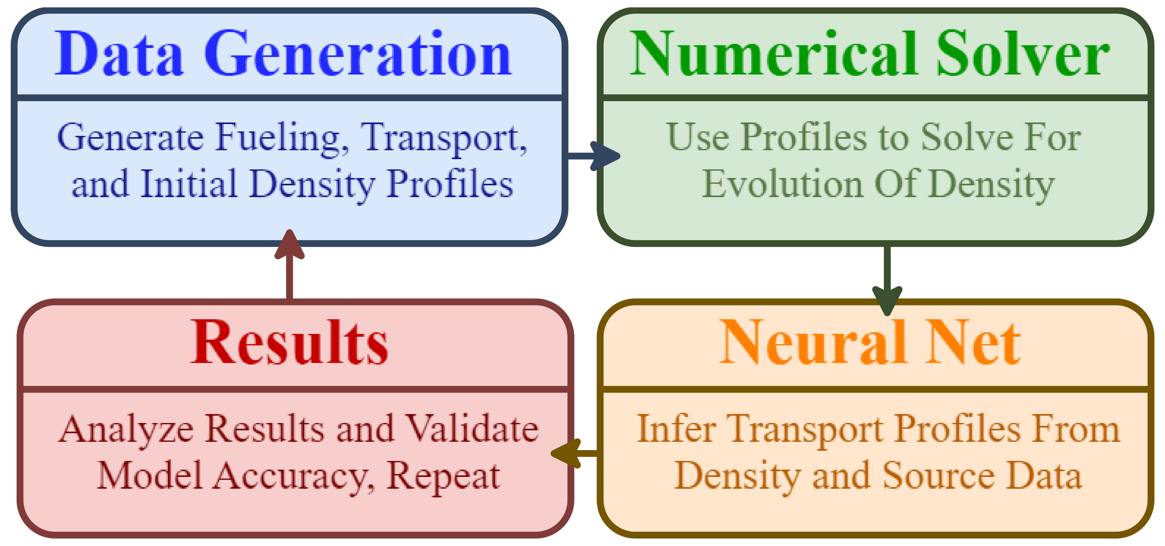
Probing and Inference of Density Pedestal Structure Through Machine Learning

J. Slone, S. Mordijck, J. Loughran
William and Mary, 200 Stadium Dr, Williamsburg, VA 23185, U.S.
jvslone@wm.edu

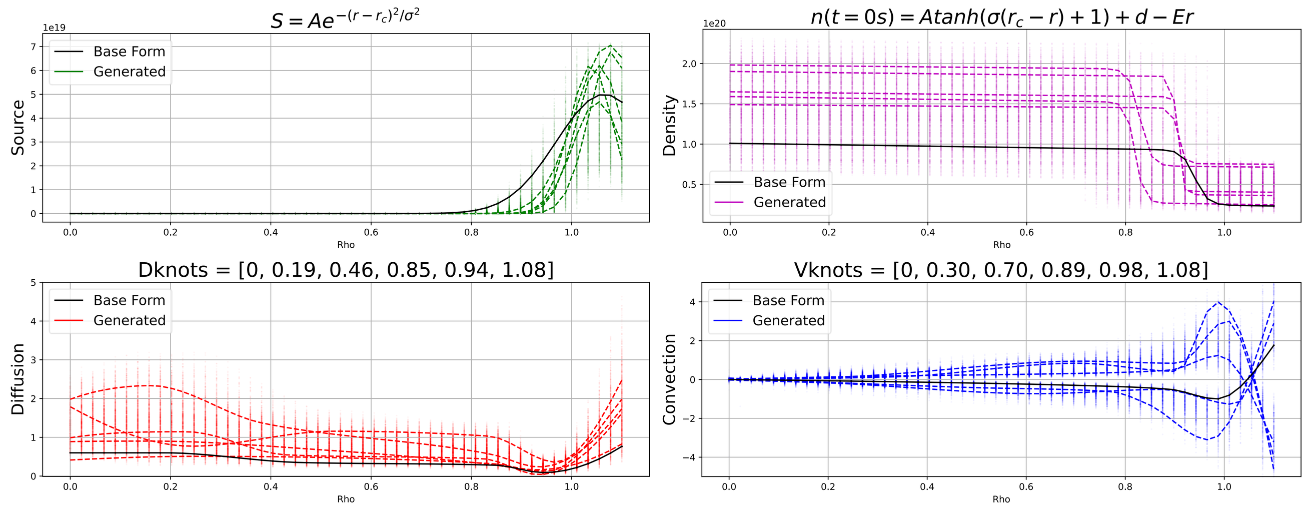
Motivation & Methods

- Understanding Density Pedestal is key to performance of high-power reactors **[1]**
- Through Machine Learning the Pedestal's structure can be probed and its contributing terms analyzed

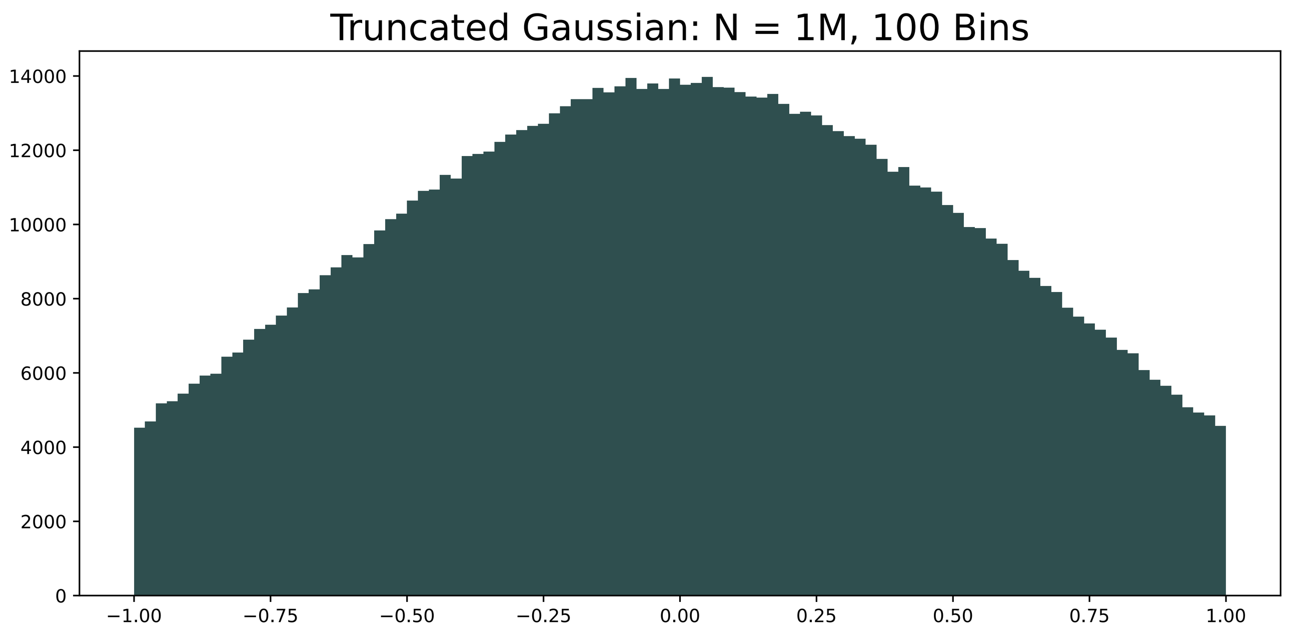


Data Generation

- Machine Learning needs **LOTS** of Data
- Wide Truncated Gaussian used for generation
- Function Parameters and Knots augmented



- 100k unique profiles for each of the above generated
- Density function distilled from [2]
- Other base forms in accordance with [3]



Accelerated Numerical Solver

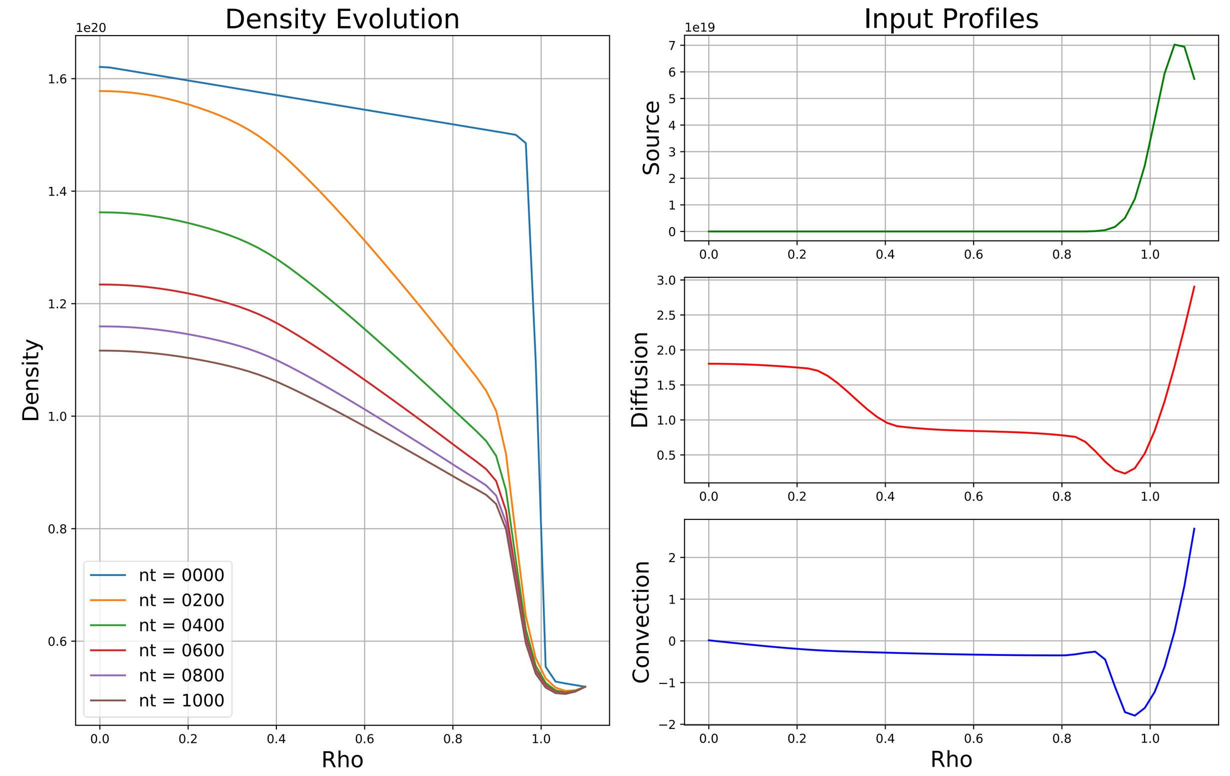
- Density modeled through Continuity and the Radial Convective-Diffusive Ansatz **[3]**:

$$\frac{\partial n}{\partial t} = -\frac{1}{r}\frac{\partial}{\partial r}(r\Gamma) + S \qquad \Gamma = -D\frac{\partial n}{\partial r} + vn$$

Boundary Conditions	Rho & Time Grid
$n(t=0s) = n_0$ $n(rho = 1.1) = C$	$rho \in 0 < rho \leq 1.1$ $nr = 50\ Steps$
$\frac{\partial n}{\partial r}(rho = 0) = 0$	$t \in 2.9s \leq t \leq 4.0s$ $nt = 1000\ Steps$

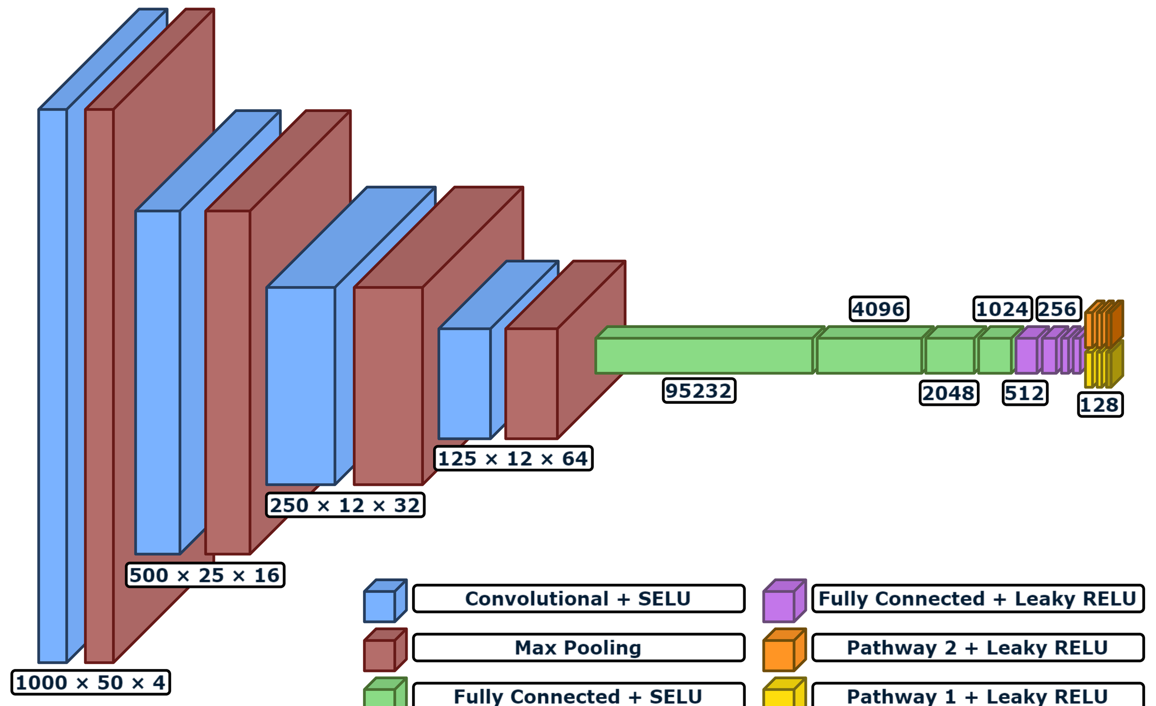
- SciPy spsolve used to compute results
- CuPy GPU Accel used to optimize run time
- 200x speedup from naïve NumPy solver

- Solver benchmarked to existing solvers within machine residual error (10^{-16})
- Near-zero rho values replaced to avoid NaN's
- Below is randomly selected profile set



Convolutional Neural Network

- Network written/trained with PyTorch & NVIDIA CUDA
- 'Compressing' network shape performs best



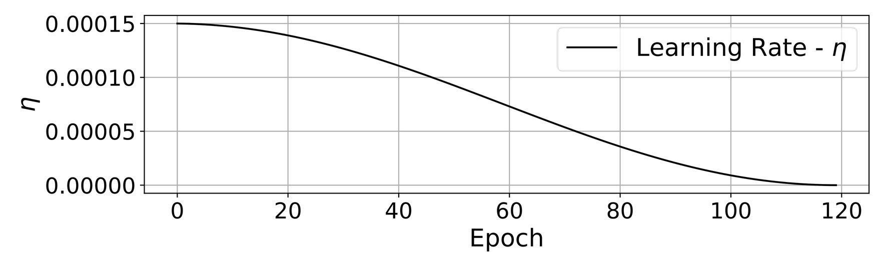
Network Inputs
Rho: 1D
Time: 1D
Source: 1D
Density: 2D

Network Outputs
Diffusion: 1D
Convection: 1D

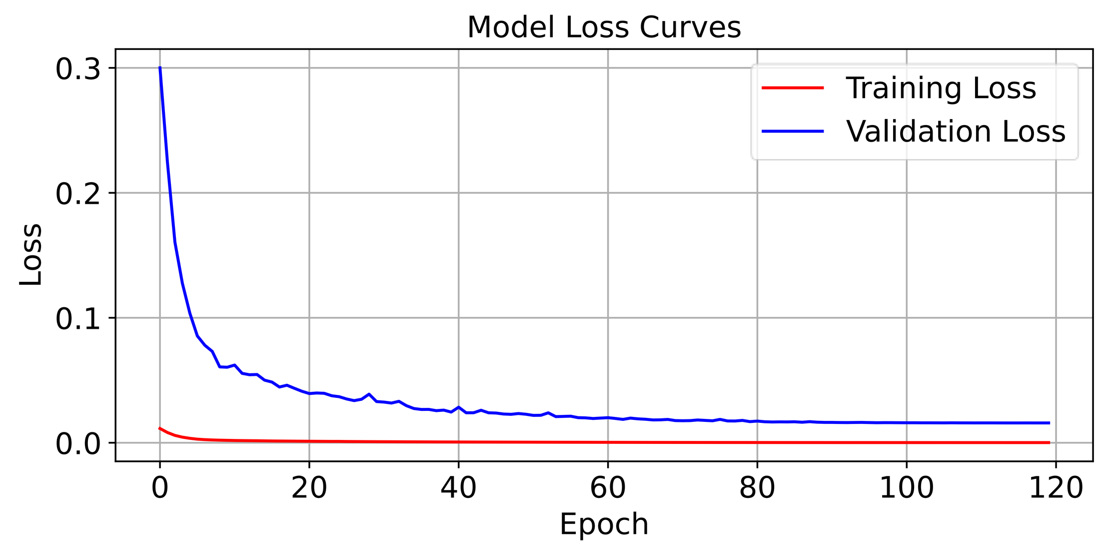
Network Characteristics

- ❑ Non-Linear Activation Functions
- ❑ Split Output Data Pathways
- ❑ Learning-Rate Scheduler
- ❑ Huberloss Criterion (delta = 0.75)
- ❑ AdamW optimizer function
- ❑ Data Standard Scaling
- ❑ Adaptive Layer Size

- Train/Val/Test split of 70/10/20, 120 Epochs
- Train Data = 100k profile sets (no repeats used)
- Model Training takes ~5hrs total (Single 4090)



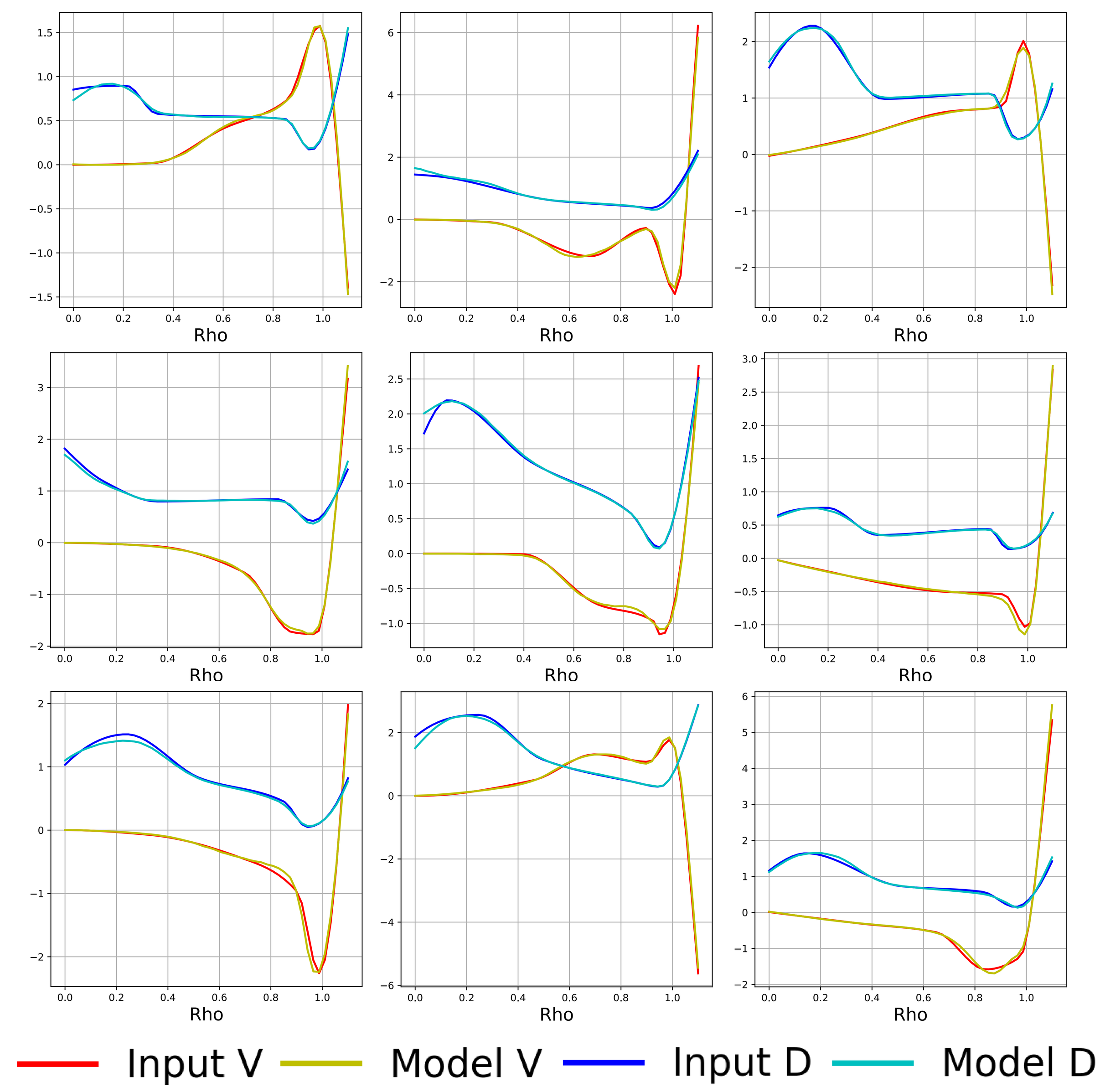
Training Loss	Validation Loss	Test Loss
0.0042	0.0159	0.0168



- Train Loss << Test Loss indicates overfitting
- Opportunity for Regularization to fix ↑

Discussion

- Examples below selected at random from test set
- Model accuracy high, with slight deviations at core
- Match to ground truth already better than experimental measurement errors



Future Work

- Validate fitted & non-fitted experimental data
- Have model infer non-constant D and V profiles
- Allow for inference in real-time with exp data
- Extract new physics from model connections
- Test inference on multiple devices for generalization

Acknowledgments

[1] S. Mordijick 2020 Nucl. Fusion 60 082006

[2] E. Stefanikova et al 2016 Rev. Sci. Instrum. 11E536

[3] A.M. Rosenthal et al 2024 Nucl. Fusion 64 036006

Work supported by US DOE under DE-FC02-04ER54698, DE-SC0019302, DE-AC02-09CH11466, DE-SC0024523 and DE-SC0014264.