Hyper Parameter Optimization

Nijat Rustamov

Introduction

An accurate understanding of transport mechanisms in nano-confined systems is critical for many applications relevant to energy and sustainability technologies. However, the intricate physics governing the gas flow in confined media challenges the scientific efforts to bridge the gap between continuum and free molecular flow scales. In an effort to mitigate this problem a highly efficient numerical simulation model powered by the lattice Boltzmann method was developed and optimized capable of undertaking extremely large and complex porous media with a wide range of Knudsen number. However, as for any other numerical method, simulation of large number of samples is restricted by computational time. In this course, I decided to start exploring and implementing what I have learned so far and apply it to the domain of my interest. In this exercise and several subsequent ones, I generate numerous artificial porous media, run the numerical simulations, and try to formulate the problem as a machine learning task to predict the flow behavior.

The numerical simulations are based on the continuous Boltzmann equation given as

$$\frac{\partial f}{\partial t} + \vec{\xi} \cdot \vec{\nabla} f + \frac{\vec{F}}{m} \cdot \vec{\nabla} f = \Omega(f) \tag{1}$$

Where f represent particle distribution function in space and left-hand side describes the movement of particles in space and time and right-hand side describes the collision dynamics. The discretized multi-relaxation time Boltzmann equation is given by

$$f_\alpha(x + ce_\alpha \delta t, t + \delta t) = f_\alpha^{eq} + \widetilde{f_\alpha} - \sum_\beta (\boldsymbol{M}^{-1}\boldsymbol{S}\boldsymbol{M})_{\alpha,\beta} \widetilde{f_\beta} + \delta t F_\alpha(x, t) \tag{2}$$

The details of what each term stands for can be found in [2]. With proper boundary conditions the method is capable of simulating flows in confined (nanoscale) media. In this work, Knudsen numbers are used to represent the scale. Knudsen number is the ratio of mean free path of the fluid to the representative pore diameter. In this work, fluid is methane gas flowing through the pores as shown in Figure 1 in Appendix. At high Knudsen numbers the flow goes into slip, transitional and free molecular flow regimes where the slip velocity cannot be neglected.

Dataset description

The dataset includes numerical simulation domains with the dimensions of 500 x 500 and the results of those simulations. Domain generation and processing was written in MATLAB and Figure 1 in Appendix section shows a few samples. The processing part included skeletonization of image, calculation of mid-axis and local pore size. Using local pore size information Knudsen number is then calculated. These parameters are used as input to the numerical simulation and the output is the x and y direction velocity distributions as well as density distribution. Numerical simulation is written in C++ from scratch using the theory of Lattice Boltzmann Method with specialized boundary conditions and equation of states to capture the physics of nano-confined gas flow. Several optimization and parallelization techniques have been applied to speed up the program and enable the utilization of HPC. For confidentiality reasons, I am unable to share the source code. Please refer to the publications in the references section for technical details [1, 2]. The Knudsen number distribution was used as input to the ML to match the x direction velocity distribution from the numerical simulation. Sample Knudsen number and velocity distributions are shown in Figure 2 in Appendix. In total, 300 samples have been generated and simulated over the period of two weeks. **To access the data used in this exercise, please follow the link to my OneDrive account, due to 25 MB limitation on GitHub I could not upload it there.** HPO_DATASET

Experimental Setup

All 300 input-output pairs are loaded and split into the training and test set. Each set was normalized by their minimum and maximum values. PCA was implemented to reduce the dimensions of the images from 500 x 500 to 15 x 10 by keeping the 98 % of the variance. The output of PCA was normalized between 0 and 1 again. Using the CNN architecture shown in Figure 3 Bayesian search was deployed to find the best settings. The search space included 4 activation functions, 4 loss functions, different number of kernels and learning rate. Details of hyper parameter optimization are given in Table 1. Keras Tuner library was used to run the optimization. Bayesian optimization was run with 75 trial points 100 epochs for each run. From the results of the search the 3 best models were selected, the summary of which is provided in Table 2. Each of the three models were trained for 500 epochs with 5-fold cross validation. The learning curves are shown in Figure 4.

Results and Discussion

Generally speaking, the results are not as great. For all 3 models, the predictions for the training and test samples are shown in Figures 5-7. The predictions of models 0 and 2 are complete nonsense. Although the color scales of true and predicted maps are not the same it is clear that the predictions are complete nonsense. Furthermore, bringing them to the same scale should not be done, because the good model predictions should match not only the general patterns but also the color scale. For model 1, training predictions are reasonable, but the test predictions are not. This could be due to the fact that the model is not able to generalize well due either to the lack of data or the over-reduction with PCA. It is worth mentioning that PCA is limited by the number of samples, hence the images are over-reduced. In the following projects I attempt down-sampling instead of dimensionality reduction.

Acknowledgements

I have used ChatGPT for debugging some parts of my Python code and in writing this report. Although it was useful in writing, debugging was not so successful, but it did point me into the correct direction for solving the small bugs in the code quickly.

Appendix

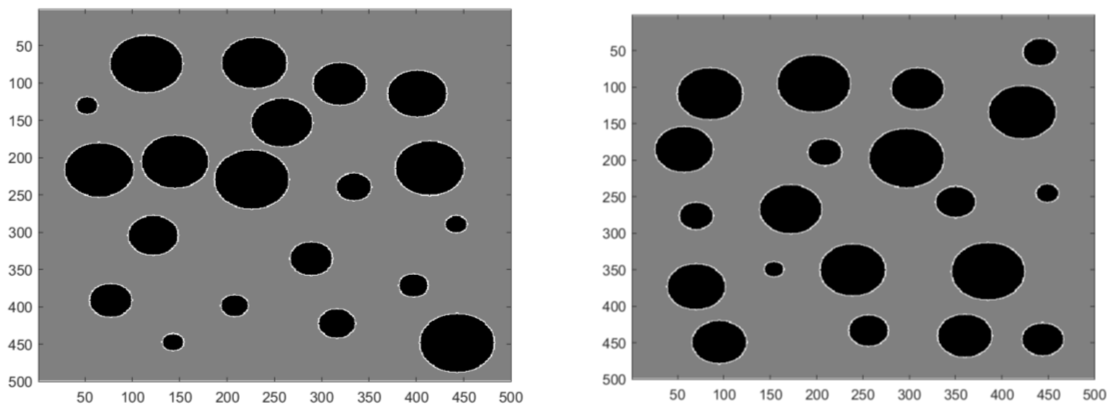

Figure 1. Randomly generated simulation domains. Gray areas indicate pores and black areas (circles) represent grains (boundaries) that fluid cannot penetrate.

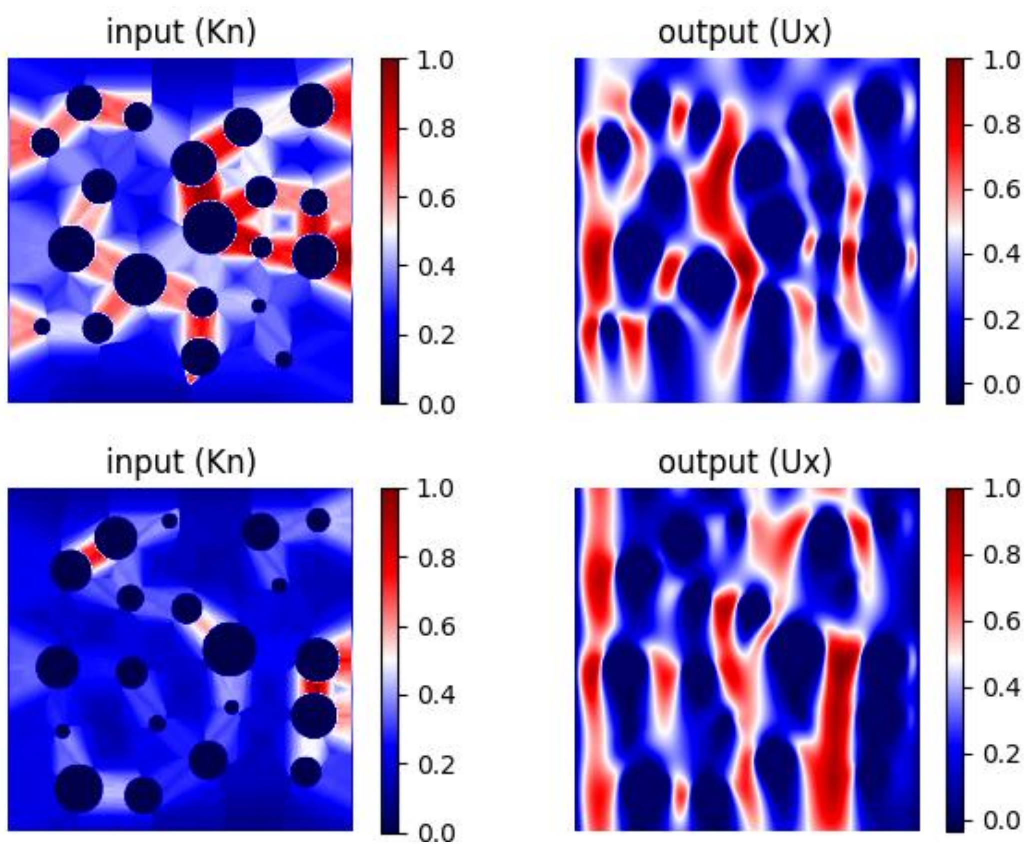

Figure 2. Sample input (left column) Knudsen number distributions and output (velocity) distribution. Data has been normalized between 0 and 1.

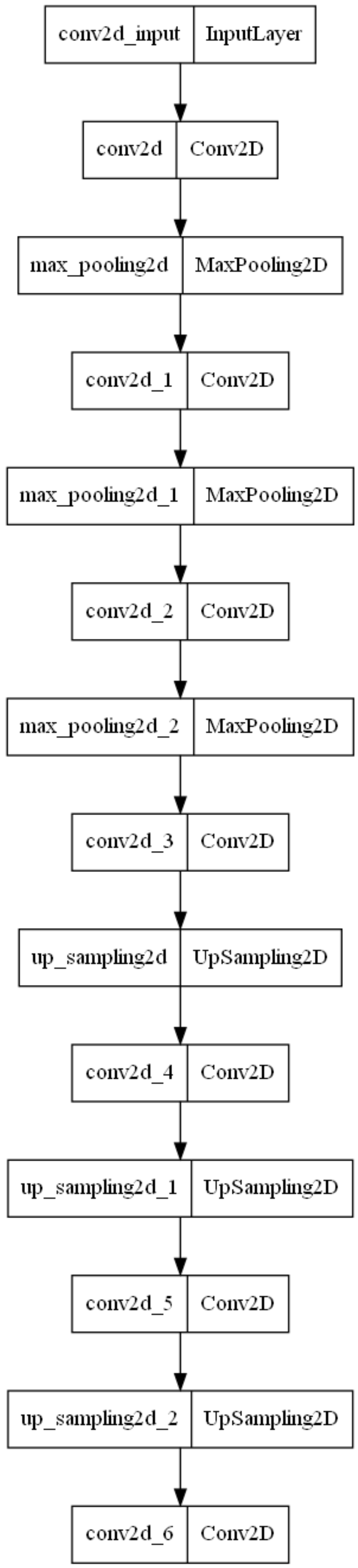

Figure 3. CNN Architecture

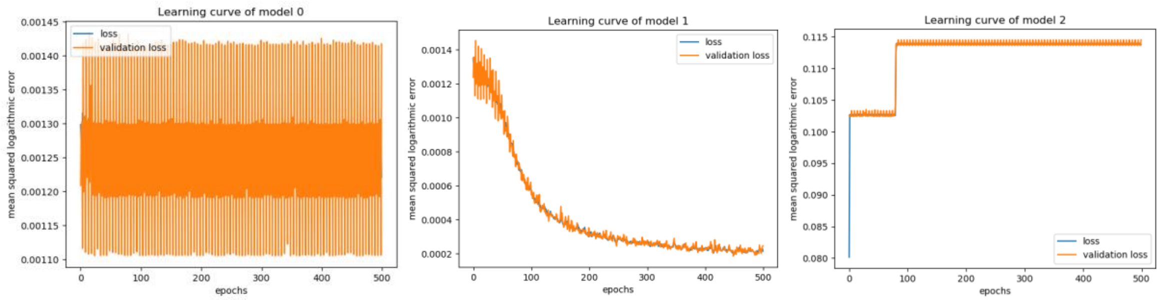

Figure 4. Training curves 3 best models

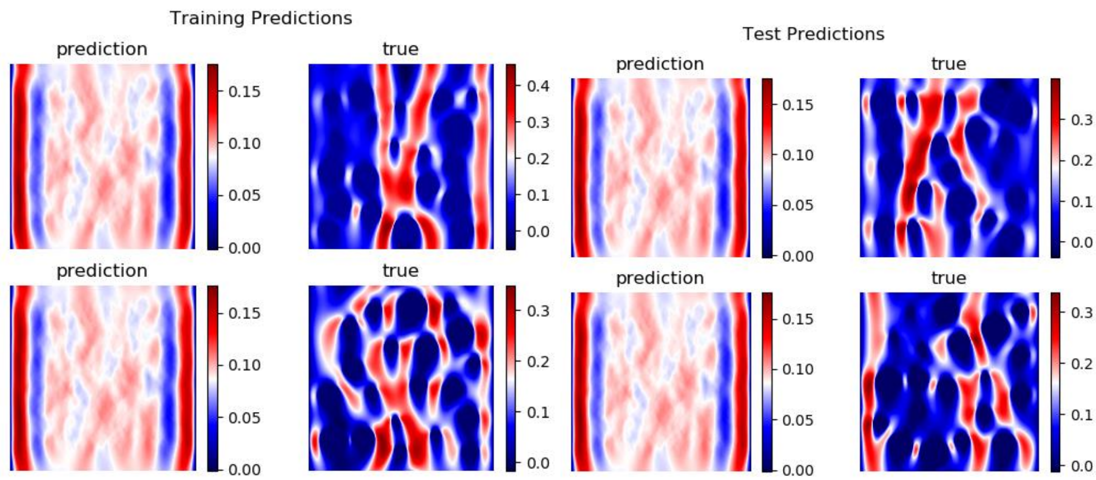

Figure 5. Model 0 predictions

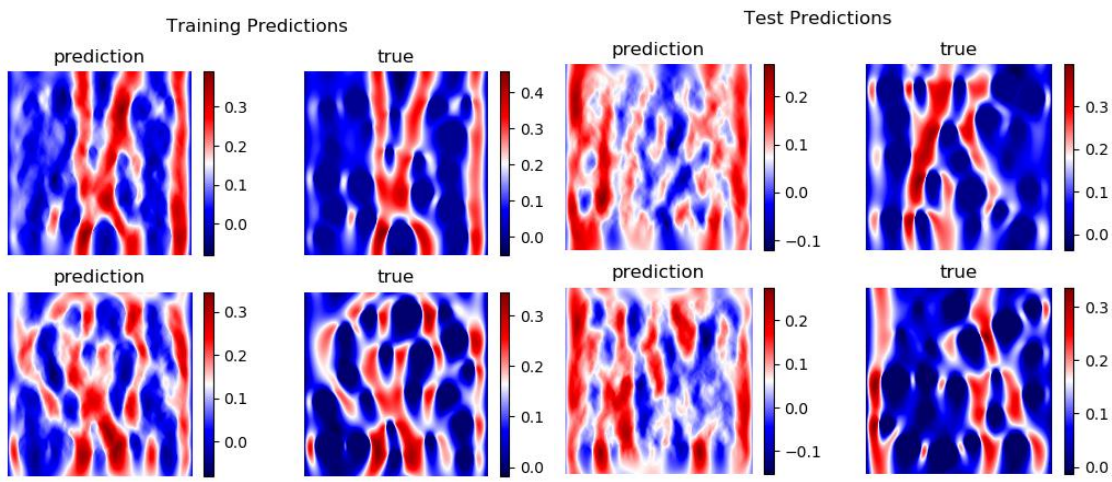

Figure 6. Model 1 predictions.

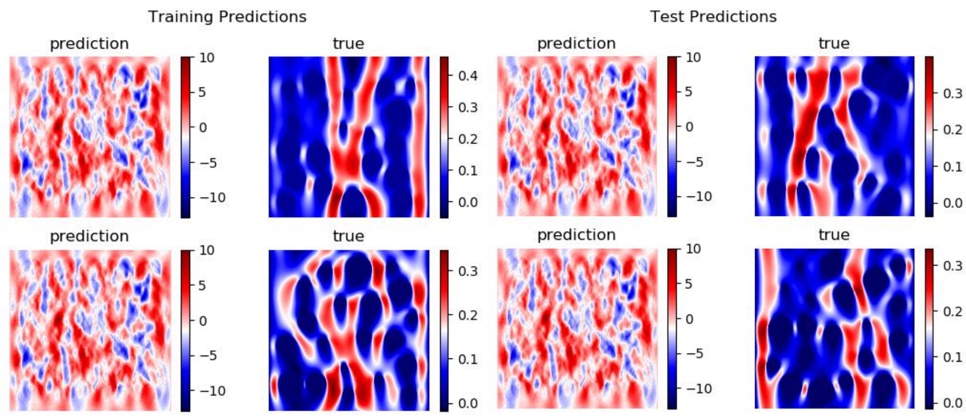

Figure 7.Model 2 predictions

Table 1. Search space for hyper-parameter optimization

Parameter	**Values**
Activation Functions	relu, sigmoid, tanh, softmax
Loss Functions	mean_squared_error, mean_absolute_error, mean_absolute_percentage_error, mean_squared_logarithmic_error
Number of Kernels	8, 16, 32
Learning Rate	1e-4 – 1e-2
Optimization Method	Bayesian Search
Optimization Setting	100 trials, 75 epochs, val_split = 0.1

Table 2. Results of Bayesian Search

Model	Parameter Setting	Validation Score	Test Score (MSE)
0	sigmoid, 16 kernels, lr = 0.0071, msle	0.00136	0.0023
1	relu, 32 kernels, lr = 0.00155, msle	0.00136	0.0038
2	tanh, 16 kernels, lr = 0.00385, msle	0.00136	1.9616

References

1. Rustamov, N., Douglas, C. C., & Aryana, S. A. (2022). Scalable simulation of pressure gradient-driven transport of rarefied gases in complex permeable media using lattice Boltzmann method. *Fluids*, *8*(1), 1. https://doi.org/10.3390/fluids8010001
2. Rustamov, N., Liu, L., & Aryana, S. A. (2023). Scalable simulation of coupled adsorption and transport of methane in confined complex porous media with density preconditioning. *Gas Science and Engineering*, *119*, 205131. https://doi.org/10.1016/j.jgsce.2023.205131