ML Pipeline 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 methods, simulation of large number of samples is restricted by the 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

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

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

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

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

300 artificial porous media were generated by randomly placing obstacles in the 500x500 domains. Then each of those domains were skeletonized to calculate local pore sizes and eventually Knudsen number distributions which are input to the numerical simulation. Numerical simulation as briefly described in the introduction section is written in C++ and is outside the scope of this work to discuss in detail. However, references [1, 2] are our publications for interested readers. 300 simulations were generated, cleaned, processed, and run over the

period of 2 weeks to generate the data. The output of the simulations are x and y direction velocity distributions as well as density distributions. For simplicity, I am only interested in x direction velocity, since only x component of driving force in the simulation is kept on. Furthermore, to gain time advantage, intermolecular forces are turned off, so the density distributions are also not considered here. In Figure 2, examples for the input and the output of numerical simulation can be found. In the end, velocity from each distribution is averaged to represent this problem as a multi-input regression problem. The distribution of these mean values across all 300 samples are shown in Figure 3.

Experimental Setup

In order to be able to run this task as a pipeline, several changes have to be made to make use of certain libraries such as auto-sklearn. The problem with most pipelines is that they are very restrictive in terms of what preprocessing can be done for the data and what algorithms can be used as estimators. So, I decided to use auto-sklearn and adjust my data to its requirements. Auto-sklearn, does not accept 3-dimensional data, so the images had been flattened to 2D. However, there are a quarter million pixels in each input image, which means it is completely useless in 2D form. For that reason, I down sampled the image to 30 by 30 and linearized so that the resulting image has less than a thousand pixels. I experimented with different sizes and found that if the image is too big the pipeline will not be able to fit anything other than random model. In addition, I tried to write my own preprocessor that includes down sampling and estimator using the examples on auto-sklearn documentation [3], but a slight change in those examples resulted in errors that I could not fix. To sum up, the input data is 300 x 900 image data, and the output is 300 mean velocity values given to auto-sklearn with 5-fold cross validation. Figure 4 illustrates a few of the down sampled images.

Results and Discussion

After running for an hour with auto-sklearn, it produced an ensemble model with the train R2 score of around 0.80. and test R2 score of around 0.60. The main constituents of this ensemble were stochastic gradient descent and gradient boosting models. Figure 5 illustrates train and test predictions on a cross plot. It is possible that if the search time was longer, it would find a model with slightly better performance, however, I have tested with a few different input sizes and runtimes, the results are not dramatically different. It even surprised me that it could even get to this point considering the problem complexity. The performance results of the model are demonstrated in Table 1.

Appendix

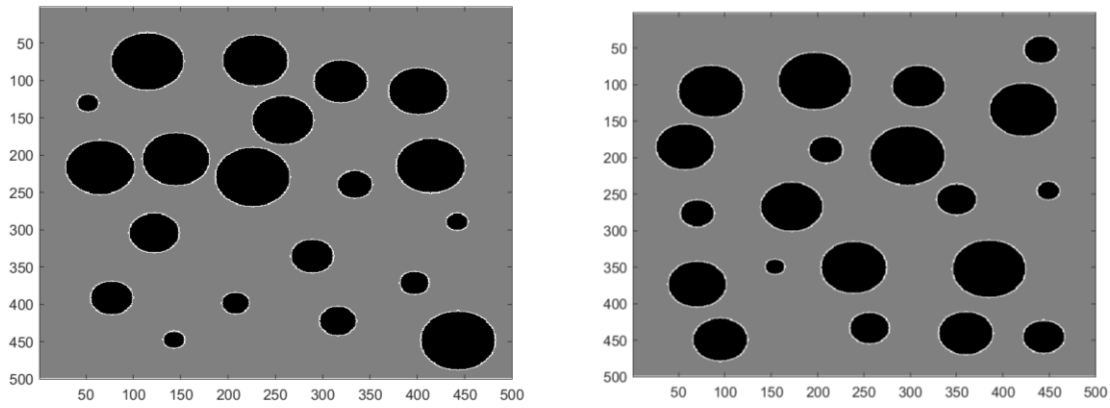

Figure 1. Randomly generated simulation domains. Gray areas indicate pores and black areas (circles) represent grains (boundaries) that fluid cannot penetrate.

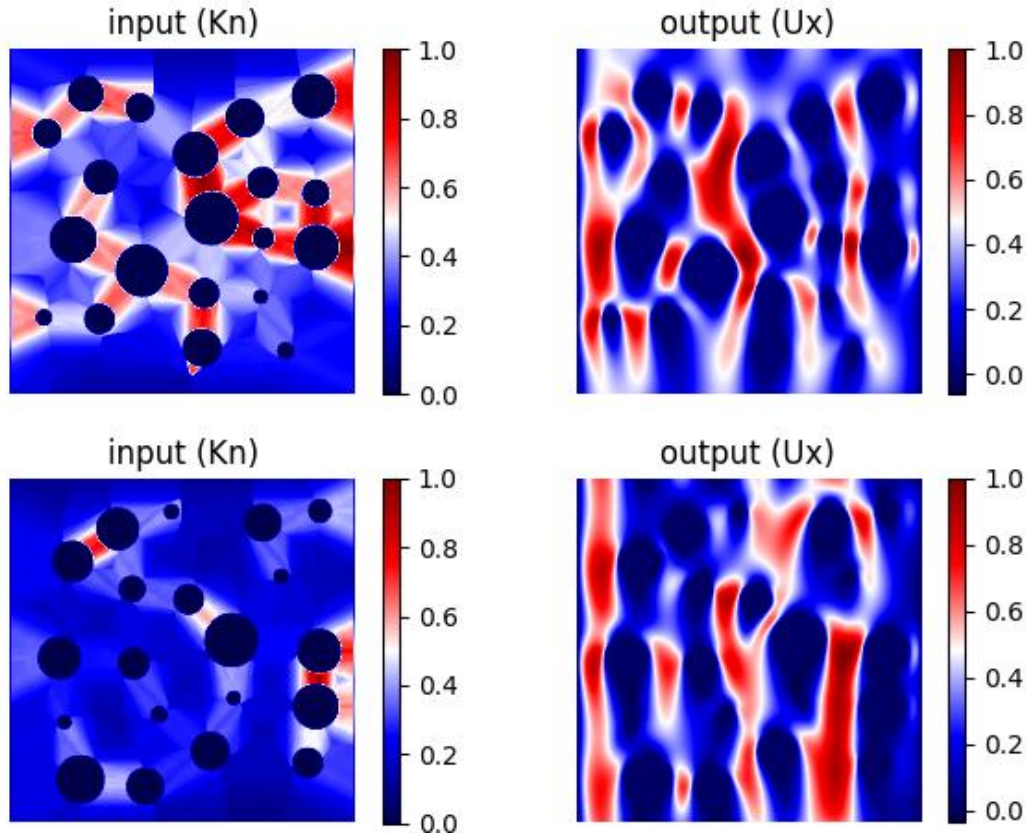

Figure 2. Sample input (left column) Knudsen number distributions and output (velocity) distribution. Data has been normalized between 0 and 1.

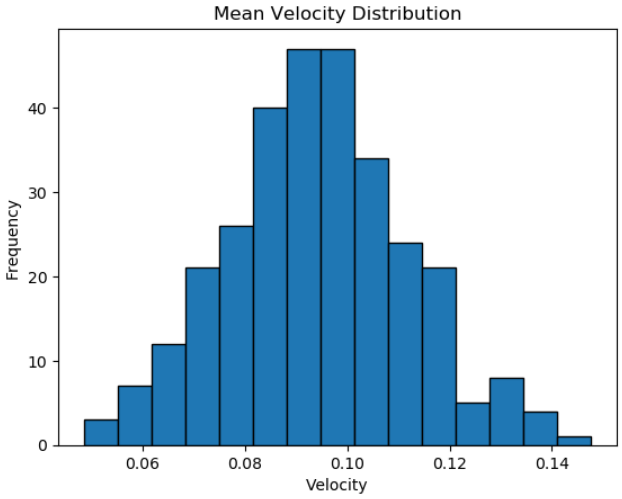

Figure 3. Mean Velocity Distribution

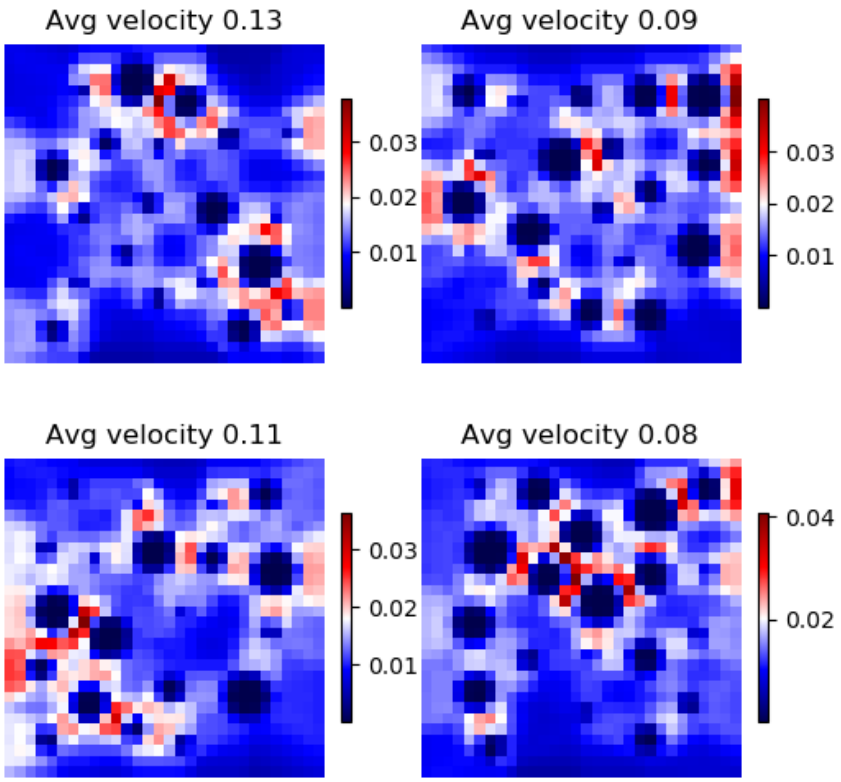

Figure 4. Down sampled input images

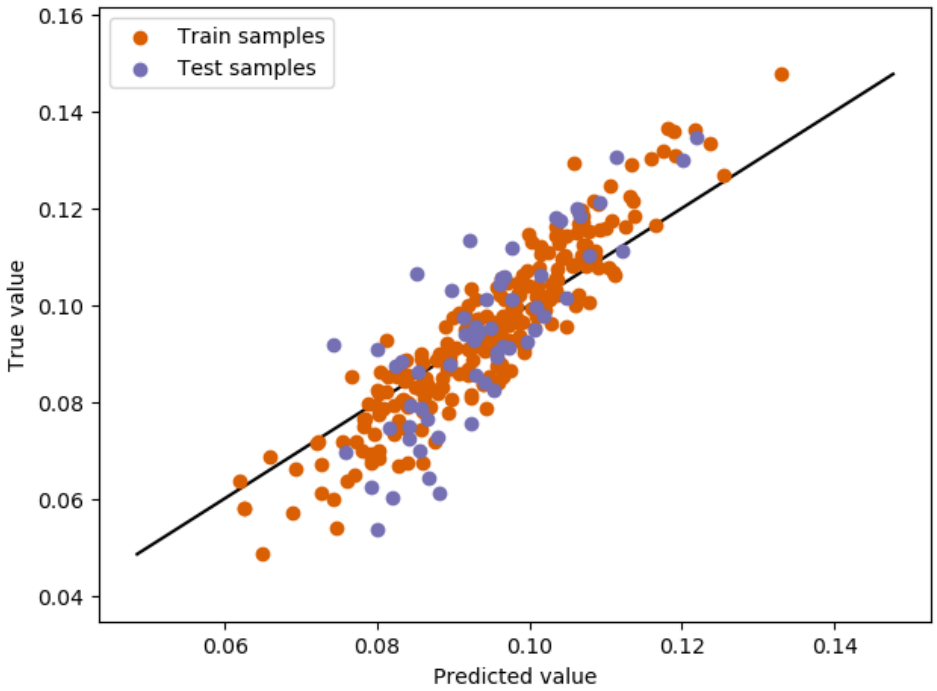

Figure 5. Auto-sklearn prediction

Model ID	Rank	Ensemble Weight	Type	Cost	Duration
115	1	0.42	Sgd	0.63	9.1
41	2	0.08	gradient_boosting	0.71	11.4
45	3	0.2	ard_regression	0.76	2.5
66	4	0.2	Sgd	0.79	2.12
111	5	0.08	gaussian_process	0.96	214.7
31	6	0.02	Adaboost	1.05	6.6

Train R2 Score: 0.81
Test R2 Score: 0.59

Table 1. Auto-sklearn performance results

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
3. https://automl.github.io/auto-sklearn/master/