Applying Machine Learning Methods to Laser Acceleration of Protons: Lessons Learned from Synthetic Data

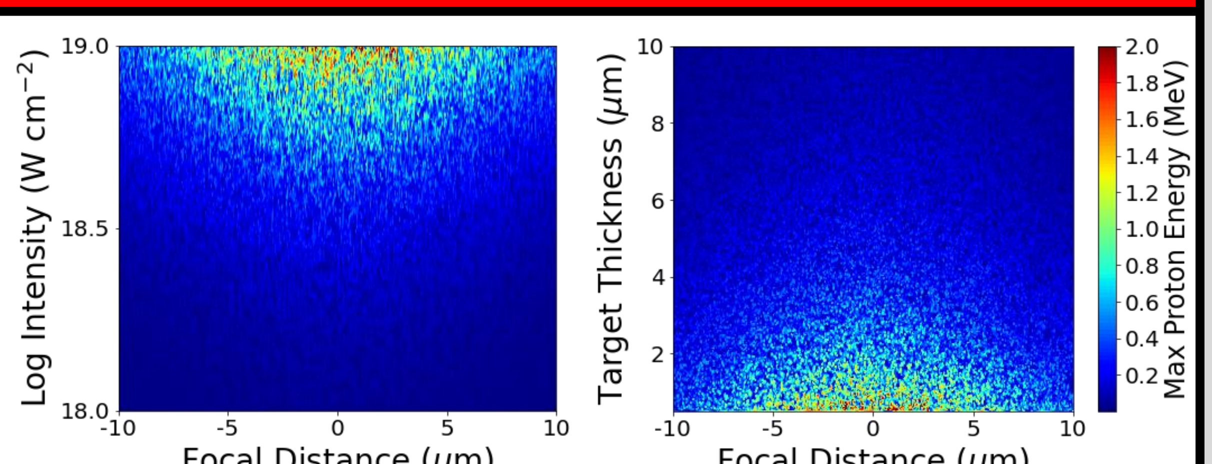
Ronak Desai[1], Thomas Zhang[1], Ricky Oropeza[1], Joseph R. Smith[2], and Chris Orban[1]
[1]Department of Physics, The Ohio State University, Columbus OH, 43210, USA
[2]Department of Physics, Marietta College, Marietta, OH, 45750, USA

Introduction

Researchers in the field of ultra-intense laser science are beginning to embrace machine learning methods for control and optimization of secondary particles and radiation [1,2,3]. In this study we consider three different machine learning methods and compare how well they can learn from a synthetic data set for proton acceleration in the Target Normal Sheath Acceleration regime that we generated using a modification of the Fuchs et al. 2005 [4] model. This allows us to compare the machine learning models to each other and to the intrinsic noise level that was added to the data. We also provide results on the computational performance and memory consumption of the machine learning methods, which are important considerations for quasi-real time operation of these methods on real experiments.

Synthetic Data

Synthetic data is generated from an analytic model by Fuchs[4] with some modifications including noise at different levels.



Fixed Inputs	Value
Wavelength	0.8 μm
Spot Size	1.5 μm
Pulse Duration	40 fs (FWHM)

Variable Inputs	Output (Energies)
Intensity	Maximum Proton
Target Thickness	Total Proton
Focal Distance	Average Proton

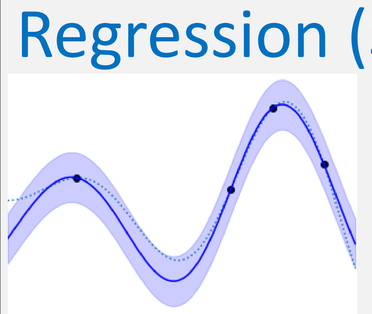
FIG 1: Distribution of Maximum Proton Energy as a function of Laser Peak Intensity and Focal Distance (left) and of Target Thickness and Focal Distance (right) from 100,000 randomly generated data points generated by the Fuchs Model. The inputs have the following ranges: Intensity ($10^{18} \rightarrow 10^{19}$ W cm^{-2}), Focal Distance ($-10 \rightarrow 10$ μm), Target Thickness ($0.5 \rightarrow 10\mu$m). The Fuchs model yields the highest energies for small target thicknesses, high intensities, and focal distances close to 0.

Note: The intensity, target thickness, and focal distance ranges were chosen to mimic the experimental ranges in Morrison et al. 2018 [5]

Learning Models

Regression

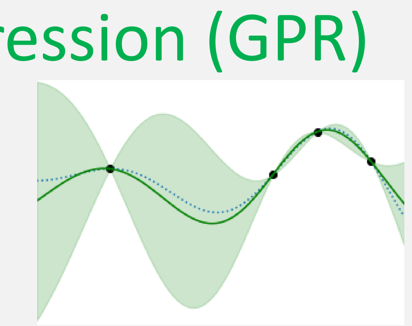
Support Vector Regression (SVR)



- **Rapids AI Library**
- Epsilon: 1e-3
- Tolerance: 1e-4
- Kernel: RBF

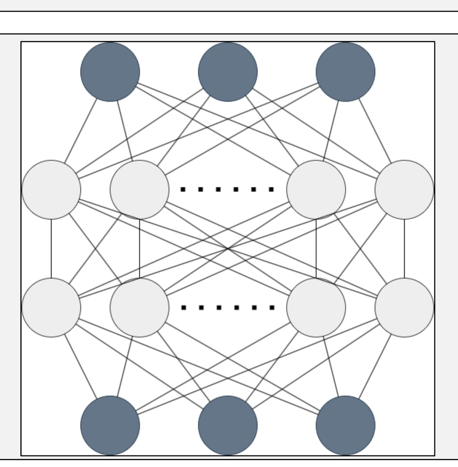
Gaussian Process Regression (GPR)

- **GPyTorch Library**
- Learning Rate: 1e-1
- Tolerance: 1e-4
- Kernel: RBF



Neural Network Model (NN)

- **PyTorch Library**
- Fully Connected
- Leaky ReLU Activation
- Adam Optimizer, 35 Epochs



Preprocessing

- Logarithmic Scaling on Intensity and Output Energies
- Standardization (Z-score) of all variables

Acknowledgements

Supercomputer allocations for this project included time from the Ohio Supercomputer Center. We also thank Alona Kryshchenko and Jack Felice for helpful conversations. We acknowledge support provided by the National Science Foundation (NSF) under Grant No. 2109222. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Results

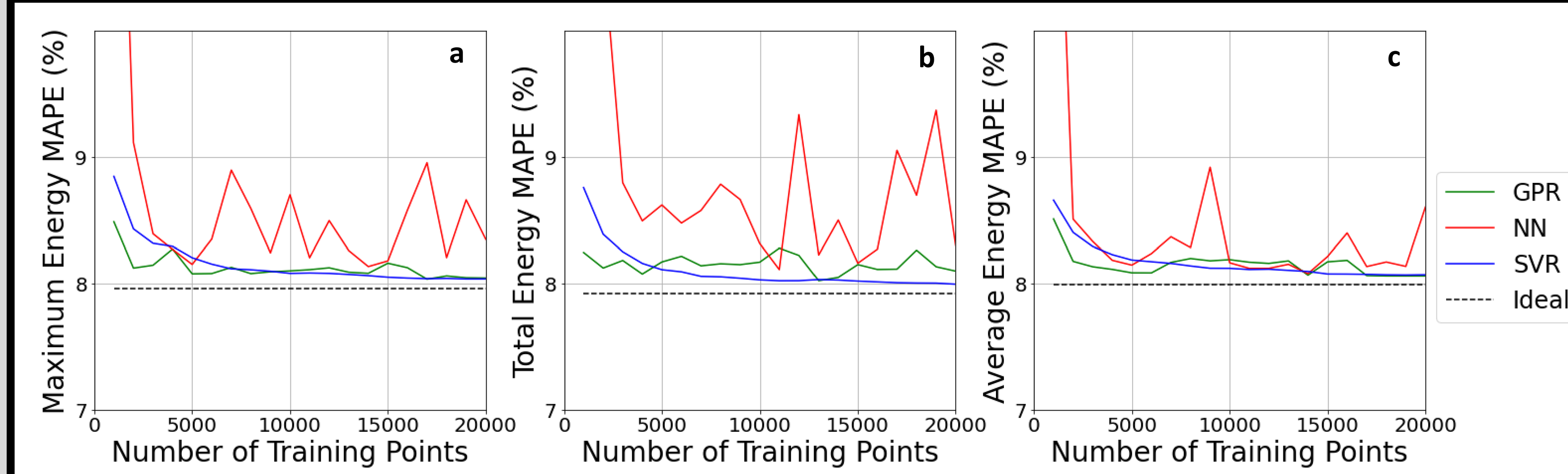


FIG 2: Testing **Mean Absolute Percentage Error (MAPE)** as a function of the number of training data points for (a) maximum proton energy, (b) total proton energy, and (c) average proton energy using a Fuchs dataset with 10% added gaussian noise. The "Ideal" dotted line is the MAPE between the noiseless and noisy models and a perfect model unaffected by the noise would reach this limit. The testing data set is kept fixed at 5,000 points here and for all the plots on this poster.

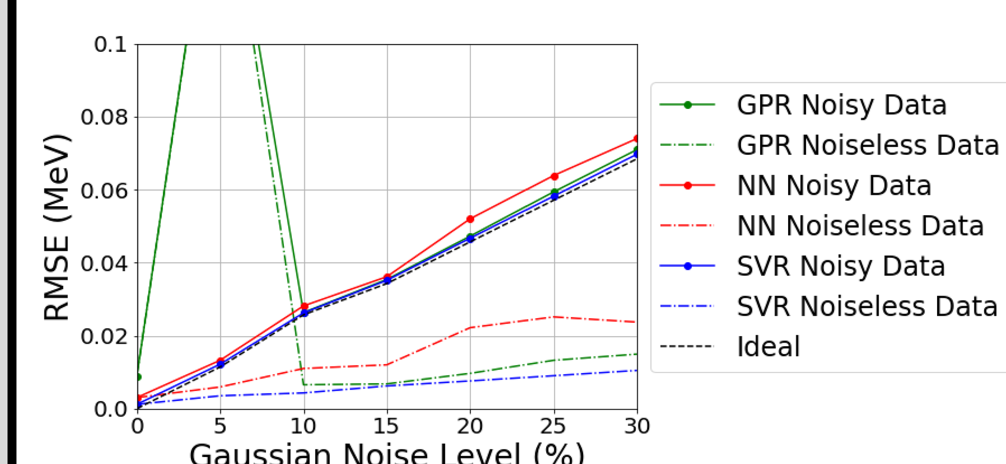


FIG 3: Root Mean Square Error (RMSE) as a function of the added gaussian noise level for a model trained on 20,000 data points for the maximum proton energy. Numerical instabilities in the GPR algorithm caused the model to perform significantly worse for the 5% noise dataset.

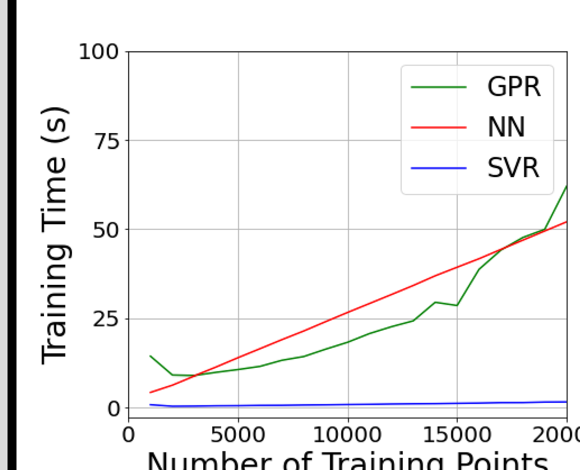


FIG 4: Total training time as a function of the number of training points for a Fuchs Data set with 10% added noise.

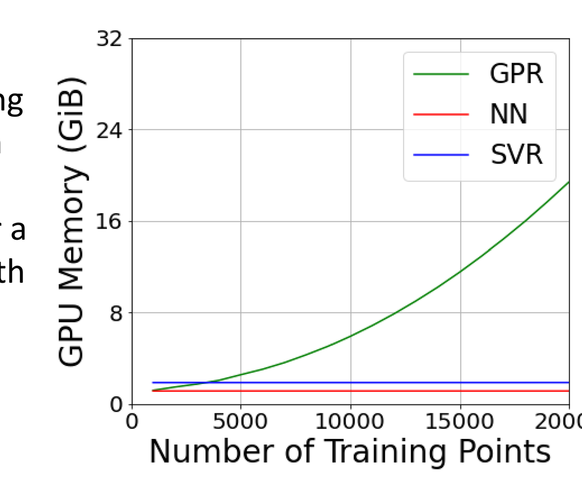


FIG 5: Amount of GPU memory utilized on average as a function of the number of training points for a given model.

Future Work

- Train models on more realistic data that explores the parameter space as a real experiment would (i.e. no random sampling of parameter space)
- Train models on ~1 million synthetic data points to prepare for experimental data from ~kHz repetition rate lasers and consider practicalities of training models in quasi-real time.
- Use trained models to predict inputs that correspond to desired outputs (i.e. optimization and control)

References

[1] Fuchs, J. et. al., Laser-driven proton scaling laws and new paths towards energy increase, Nature Physics 2 (2005).
[2] T. Ma, D. Mariscal, R. Anirudh, T. Bremer, B. Z. Djordjevic, T. Galvin, E. Grace, S. Herriot, S. Jacobs, B. Kailkhura, et al., Plasma Physics and Controlled Fusion 63, 104003 (2021).
[3] A. Dopp, C. Eberle, S. Howard, F. Irshad, J. Lin, and M. Streeter, High Power Laser Science and Engineering p. 1–50 (2023).
[4] Fuchs, J. et. al., Laser-driven proton scaling laws and new paths towards energy increase Nature Physics 2 (2005).
[5] Morrison, J. et. al, MeV proton acceleration at kHz repetition rate from ultra-intense laser liquid interaction, New Journal of Physics 20 (2018).

THE OHIO STATE UNIVERSITY