Appendices for "Neural Network based Prediction on Equation of State with Physical Constraints"

Dongyang Kuang[1,2,*], Shiwei Li[3], Buxuan Wang[2], Chao Xiong[3,*], Shichang Zhang[3], and Yanyao Zhang[4,*]

[1]Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai), Zhuhai, Guangdong 519082, CHINA
[2]School of Mathematics (Zhuhai), Sun Yat-sen University, Zhuhai, Guangdong 519082, CHINA
[3]School of Digital Media Engineering and Humanities, Hunan University of Technology and Business, Changsha, Hunan 410205, CHINA
[4]Earth and Planetary Sciences, Stanford University, Stanford, California 94305, USA
[*] *Corresponding authors:*
yanyaozh@stanford.edu(Yanyao Zhang), xiongchao@hutb.edu.cn(Chao Xiong), kuangdy@mail.sysu.edu.cn(Dongyang Kuang)

ABSTRACT

The equation of state (EOS) is essential for understanding material behavior under different pressure-temperature-volume (P-T-V) conditions across various disciplines. Traditional models, such as the Mie-Grüneisen-Debye equation, rely on thermodynamic assumptions and expert knowledge, while classical Gaussian process based machine learning approaches can be sensitive to choice of kernels and are limited by scalability and extrapolability. To overcome these limitations, we propose a neural network based physics informed deep learning method (EOSNN) that jointly learns multiple EOS surfaces from diverse data sources, including static and dynamic compression and *ab initio* calculations. Additionally, a probabilistic model is developed to account for both aleatoric and epistemic uncertainties. Our numerical experiments show that EOSNN outperforms traditional and Gaussian process methods in several aspects including accuracy, robustness and extensibility. Particularly on the challenging partially supervised task where energy information is limited on part of the Hugoniot curve, our method's prediction for the energy off-Hugoniot can still reach a R^2 score as high as 0.83, correlation coefficient greater than 0.95 and RMSE as low as 0.52 eV/atom with proper selection of regularization for physical consistency. This result is even slightly better than the fully supervised case for traditional regression method based on the Mie-Grüneisen equation. These benefits can be further enhanced with physics-informed regularizations on quantities such as heat capacity (C_V), Grüneisen parameter (γ) or bulk modulus (K_T).

Appendix A: Details on the Network Design and training algorithms

P-Net is designed to approximate surface $P = P(V,T)$ via the form $f(V)g(T)+h(V)$ where unknown free-form functions f, g and h each is approximated by a mutlti-linear perceptron (MLP) with a single hidden layer of 64 hidden units. Similarly, E-Net also takes the form $f^*(V)g^*(P)+h^*(V)$ for the approximation of surface $E = E(V,P)$. In our experiments, designs like this can help produce smoother surface. An exception is for the case where regular regression (such as polynomial regression) is performed first and neural network is used for fitting the remaining residual (denoted by PR+NN in our manuscript). In such cases, both P-Net and E-Net are chosen to be just simple MLPs with a single hidden layer containing 32 hidden units. The actual training algorithms for the deterministic model and the probabilistic model are list below separately as **Algorithm** 1 and **Algorithm** 2. Other hyperparameters for reproducing reported result of this work can be checked at our Github repository: https://github.com/dykuang/DL4EOS.

Algorithm 1 Training algorithm for the deterministic model

Require:

• Data (V^S, T^S, P^S) from static experiments, data (V^H, P^H) along Hugoniot curve.
• Maximum epochs N, Error tolerance ε, Hyper parameter α
• Choice of networks $f_P(V,T;\theta)$ and $f_E(V,P;w)$
while $n < N$ and loss $> \varepsilon$ **do** *# Other stopping criterion can be used.*
    # —- forward pass —-
    Calculate $Loss_P = \sum_{i=1}^{N} ||f_P(V_i^D, T_i^D; \theta) - P_i^D||_2$
    Calculate $Loss_E = \sum_{i=1}^{M} ||f_E(V^H, P^H; w) - E_i^H||_2$
    Calculate losses for physical constraints:

$$Loss_C = L^+(\frac{\partial E}{\partial T}|_V) + L^+(\frac{\partial P}{\partial T}|_V) + L^-(\frac{\partial E}{\partial V}|_T) + L^-(\frac{\partial P}{\partial V}|_T)$$

    Compute the total loss: $Loss_{total} = Loss_P + Loss_E + \alpha Loss_C$
    # —- backward pass —-
    Update network weights via an optimizer (e.g. Adam[1]).
end while
Save the model with optimum weights $f_P(V,T;\theta^*)$ and $f_E(V,P;w^*)$

Algorithm 2 Training algorithm for the probability model

Require:

• Data (V^S, T^S, P^S) generated from static experiments and data (V^H, P^H) generated from Hugoniot experiment.
• Maximum epochs N, Error tolerance ε, Hyper parameter α
• Choice of networks $f_P(V,T;\theta)$ and $f_E(V,P;w)$ (with last layers replaced by a Gaussian distribution layer.)
while $n < N$ and loss $> \varepsilon$ **do** *# Other stopping criterion can be used.*
    # —- forward pass —-
    Calculate $Loss_P = -NLL(f_P(V^D, T^D; \theta), P^D)$
    Calculate $Loss_E = -NLL(f_E(V^H, P^H; w), E^H)$
    Calculate losses for physical constraints:

$$Loss_C = L^+(\frac{\partial \bar{E}}{\partial T}|_V) + L^+(\frac{\partial \bar{P}}{\partial T}|_V) + L^-(\frac{\partial \bar{E}}{\partial V}|_T) + L^-(\frac{\partial \bar{P}}{\partial V}|_T)$$

    Compute the total loss: $Loss_{total} = Loss_P + Loss_E + \alpha Loss_C$
    # —- backward pass —-
    Update network weights via an optimizer.
end while
Save the model with optimum weights $f_P(V,T;\theta^*)$ and $f_E(V,P;w^*)$

Appendix B: Prediction of pressure-volume-temperature (P,V,T) surface $P = P(V,T)$ using traditional semi-empirical/empirical equations.

In 5-fold cross-validation (5CV) experiments, 20 (P,V,T) data points were randomly divided into a training set with 16 data points and a testing set with 4 data points in each experiment. The 16 training data were fitted with the EOSfit7[2] software using the Mie-Grüneisen-Debye equation[3] coupled with both Birch-Murnaghan and Vinet equations[2] A least-square fitting method was used with relaxed parameters of $V_0, K_0, K_0', \theta_0, \gamma$ and q. In fold4, EOSfit7 provides γ with a negative value, and thus, we fixed γ to 1.1 in the fitting. The obtained equation of state (EOS) parameters are listed in **Supplementary Table D1**. These parameters were further used to calculate the pressure for the 4 testing data at corresponding (V, T) conditions. The comparison of predicted and *ab initio* pressure values is listed in **Supplementary Table D2**.

In the experiment where data under low P-T condition is used as training and the rest as testing, a similar regression process was conducted using the EOSfit7 software[2]. In region I, EOSfit7 gives negative values for q as a result of optimization, thus we fixed q to 0.5 in the fitting. Obtained thermodynamic parameters are listed in **Supplementary Table D4** and predicted pressure values at the rest P-T conditions are listed in **Supplementary Table D5**.

Appendix C: Prediction of internal energy (E) surface using the Mie-Grüneisen equation

In the 5-fold cross-validation (5CV) experiments presented in **Section "Supervised Joint Learning with P-V-T-E Data"**, a least-square fitting method was adopted to fit the 16 training data using the Mie-Grüneisen equation,

$$P(V,E) = P_0(V) + \left(\frac{\gamma}{V}\right)[E(V) - E_0(V)] \tag{C1}$$

where P_0, E_0 and V_0 are the reference pressure, energy and volume, are fixed parameters with their values being the $E_0 = -8.9431$ eV/atom, $P_0 = 12$ GPa, $V_0 = 5.6684$ Å^3/atom[4]. γ is the Mie-Grüneisen parameter[3], γ is typically assumed to be

$$\gamma = \gamma_R \left(\frac{V}{V_0}\right)^q \tag{C2}$$

where γ_R is γ at ambient conditions and q is a volume-independent constant[3]. with learned parameters in (C2), E values can then be solved from equation (C1).

In the experiment where training data are from the region with lower P-T condition and the rest as testing, a similar regression process was conducted using equations (C1) and (C2).

In the partially supervised joint learning task with P-V-T data from static experiments and P-V paired gathered along Hugoniot as demonstrated in **Section Partially Supervised Learning with Hugoniot**, we did not find a representative traditional method that does similar jobs to our best knowledge. Instead as a baseline for comparison, we adopted the 25 Hugoniot (P^H,V^H) and used equation (C3) which is based on the conservation of energy[3] to calculate E^H on Hugoniot first:

$$E^H = E_R + \frac{1}{2}(P^H + P_R)(V_R + V^H) \tag{C3}$$

The resulting (P^H,V^H,E^H) are then combined with equation (C1) and (C2) for estimating proper γ_R and q. Once these two parameters are learned ($\gamma_R = 10.67, q = 2.56$ with the given data), the target energy E off Hugoniot are then again solved from (C1) with the 20 (P,V,T) from previous *ab initio* calculations[5].

Appendix D: Supplementary Tables for Numerical Experiments in the Main Manuscript

Detailed tables are uploaded as '***Supplement Table.xlsx***' files during submission along with the main manuscript which includes:

- **Supplementary Table D1**. Equation of state parameters learned for each fold in the 5-fold cross-validation (5CV) experiments.
- **Supplementary Table D2**. Predicted pressure values by semi-empirical/ empirical equations of state in the 5-fold cross-validation (5CV) experiments.
- **Supplementary Table D3**. Root mean square error of pressure (RMSE-P) and energy (RMSE-E) in 5-fold cross-validation (5CV) experiments.
- **Supplementary Table D4**. Equation of state parameters used in the experiments with low P-T data as training data and the rest P-T data as testing data.
- **Supplementary Table D5**. Predicted pressure values by semi-empirical/ empirical equations of state with low P-T data as training data and the rest P-T data as testing data.
- **Supplementary Table D6**. Root mean square error of pressure (RMSE-P) and energy (RMSE-E) in the experiments with low P-T data as training data and the rest P-T data as testing data.

Supplementary Tables D1 to D3 show the experimental data of ***Figure 3*** in the main manuscript and Tables D4 to D6 show the experimental data of ***Figure 4*** in the main manuscript. The RMSE used in Tables D3 and D6 is defined as:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2} \tag{D1}$$

where n represents the number of folds, y_i denotes the true value for the i^{th} data, and $\hat{y}_i$ indicates the predicted value for the i^{th} data.

In supplementary Table D1, the "Mean" is the average of the RMSE values of the 5CV experiments. The standard deviation (Std.Dev.) was calculated using the following formula:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\mu)^2} \tag{D2}$$

where n represents the number of folds, y_i denotes the RMSE values for the i^{th} data, and μ indicates the "Mean".

Appendix E: Supplementary Figures from Numerical Experiments in the Main Manuscript

Here we demonstrate some selected figures from numerical experiments conducted in the main manuscript for better supporting observations and conclusions made in the experiment section.

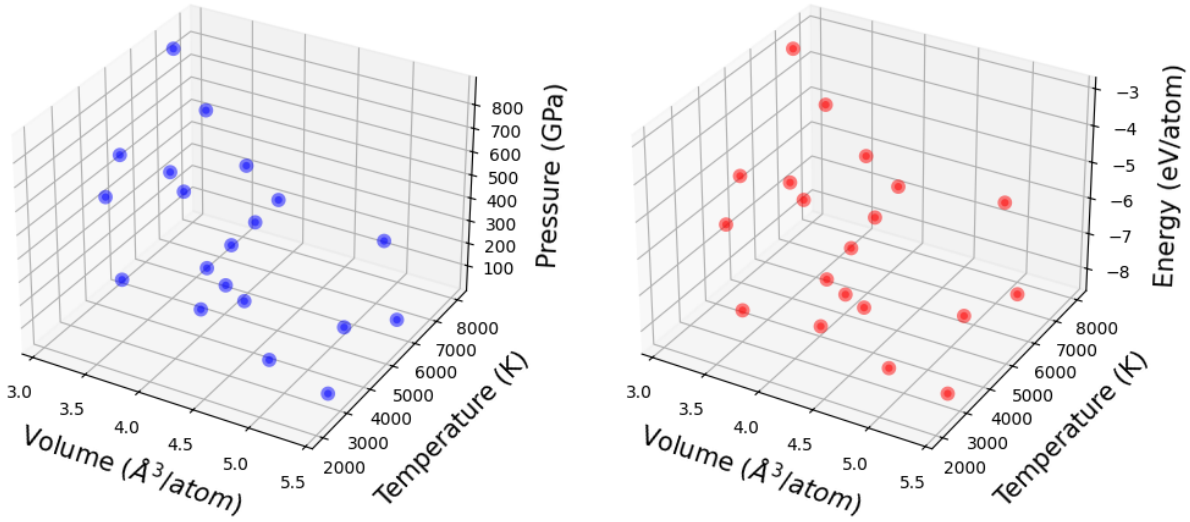

Figure E1. A visualization of 20 *P-V-T-E* data points used in the supervised task as in Section "Supervised Joint Learning with *P-V-T-E* Data".

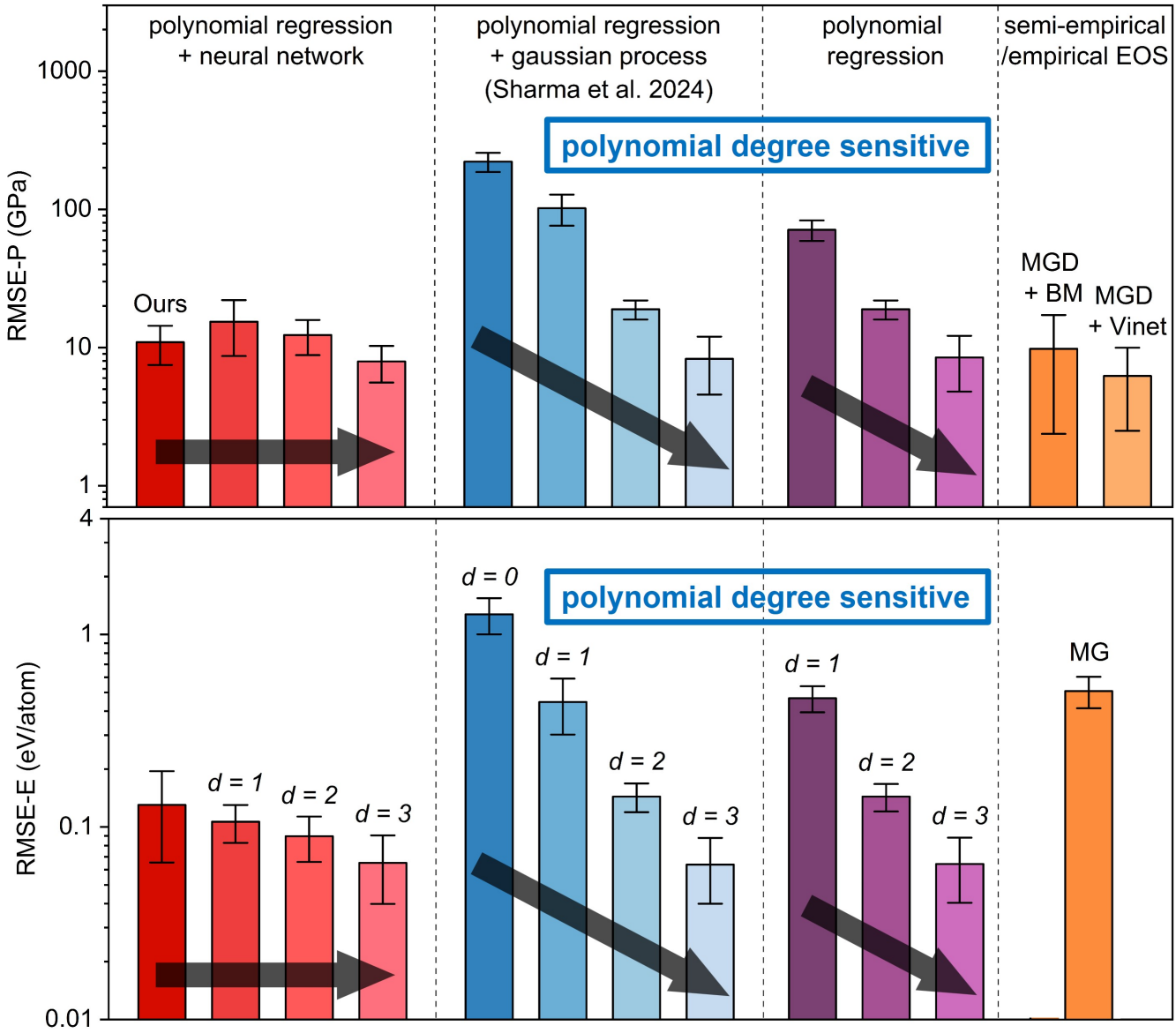

Figure E2. A more detailed comparison about robustness on key parameters when EOSNN is compared with other methods. As seen from the figure, the performance of classical machine learning methods such as Gaussian processes or regular polynomial regressions rely heavily on the choice of polynomial degree. Classical physical methods require extensive expertise on the selection of physical formula. EOSNN demonstrates more robustness w.r.t. choice of polynomial degree (if polynomial regression is used as the base predictor) and does not require particular expertise for selecting proper physical formulas.

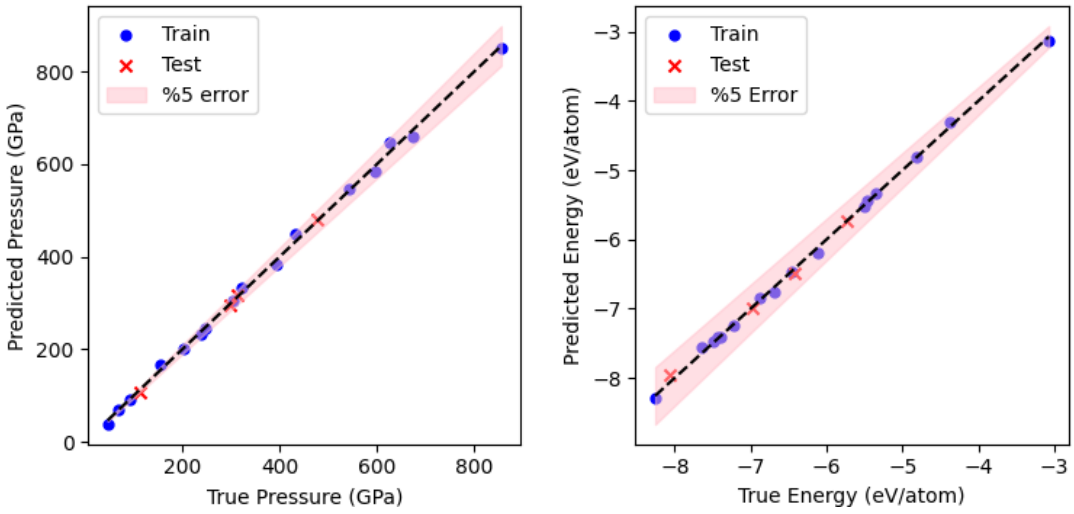

Figure E3. The prediction from fold 1 against the true values with the proposed method in the supervised experiment with *P-V-T-E* data from Section "Supervised Joint Learning with *P-V-T-E* Data".

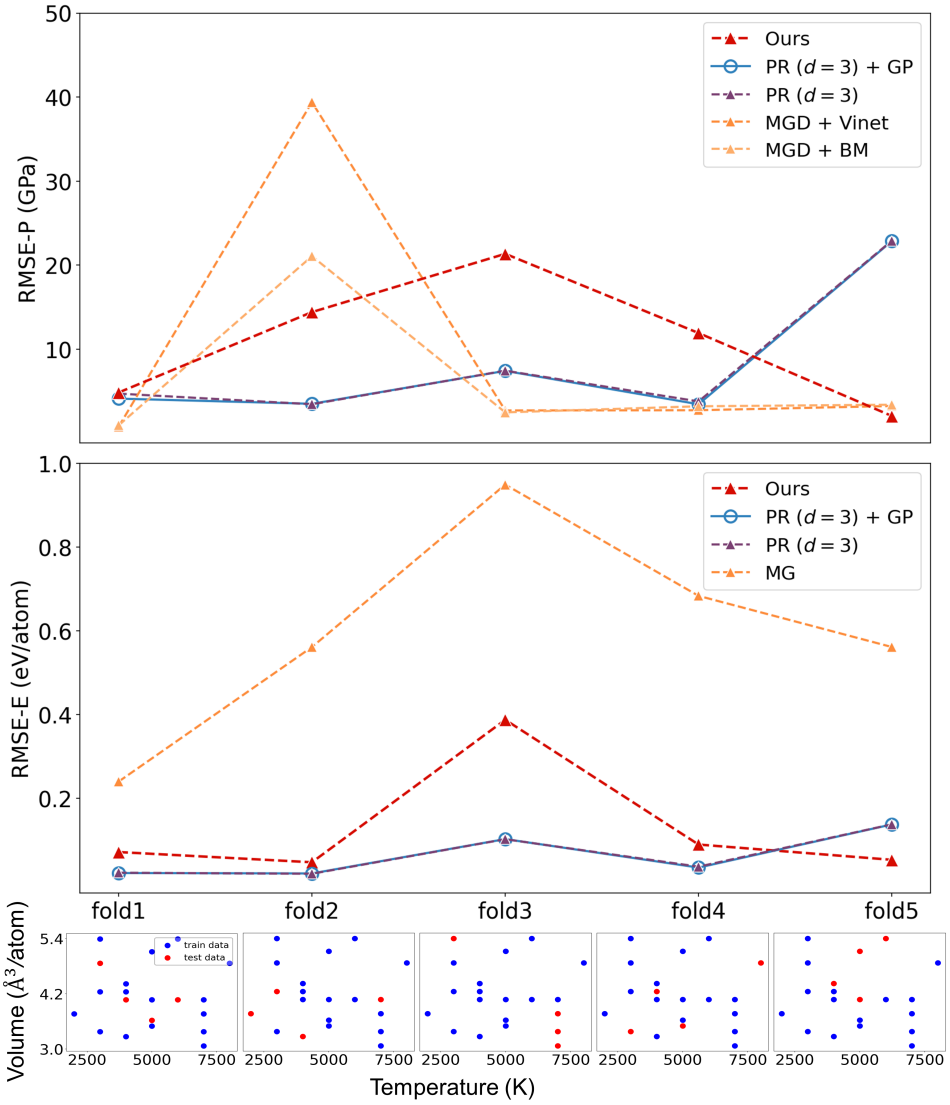

Figure E4. RMSE calculated on the test set among different models on the pressure prediction (top row) and energy prediction (middle row)in the supervised experiment with *P-V-T-E* data from Section Supervised Joint Learning with *P-V-T-E* Data. Bottom row: train/test data points as in different fold (blue for data used in training and red for test).

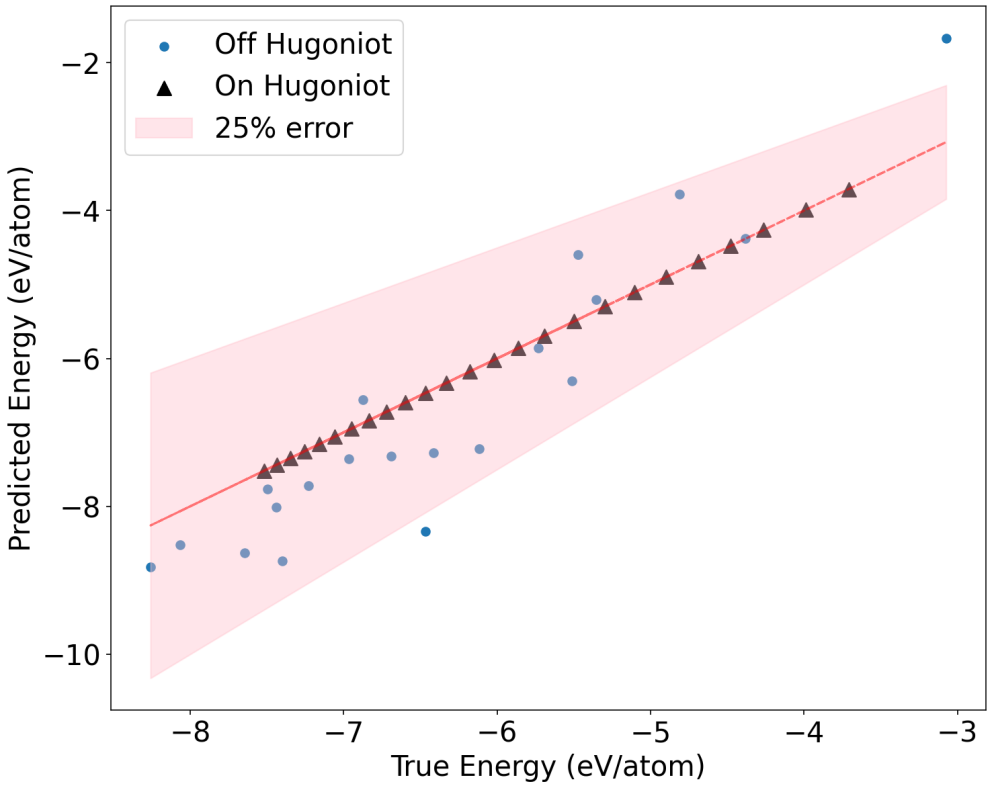

Figure E5. The prediction of energy using Mie-Grüneisen EOS as in the experiment from Section "Partially Supervised Learning with Hugoniot".

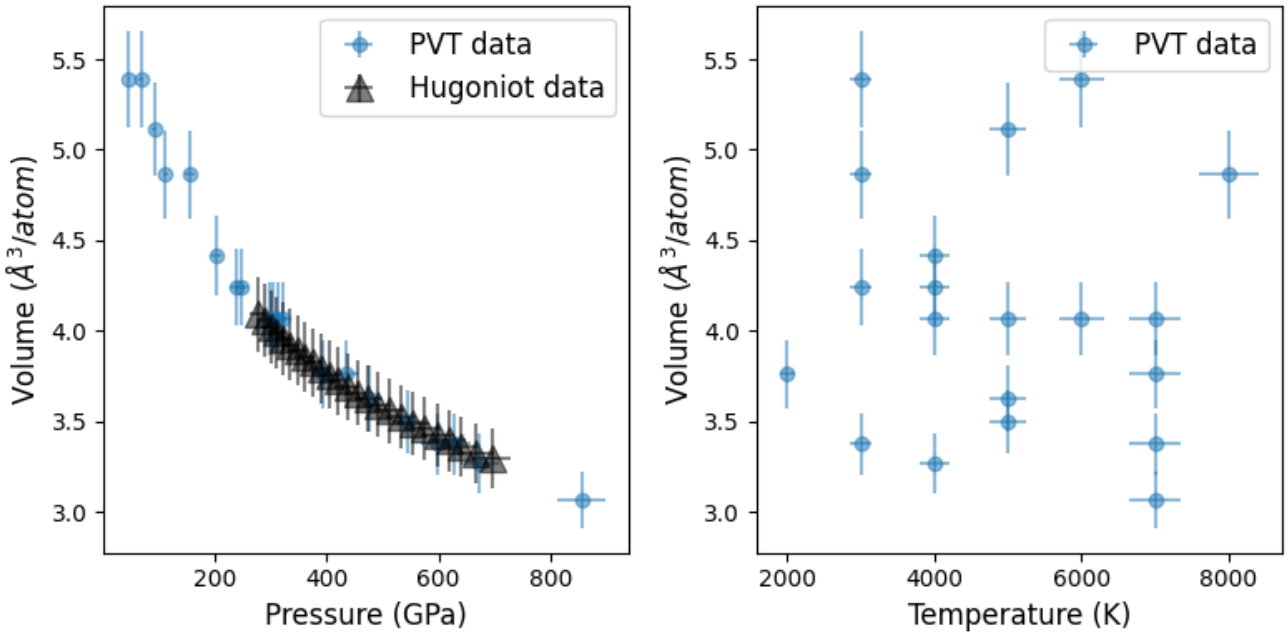

Figure E6. Noisy data accounting for aleatoric uncertainty used in Section "Probabilistic Model with Uncertainty Quantification". Left: (P^H, V^H) from Hugoniot experiment, plotted in the same P-V plane as the 20 PVT data. Right: 20 PVT data on the T-V plane. Both mean values and the standard deviations σ used to sample training data are shown.

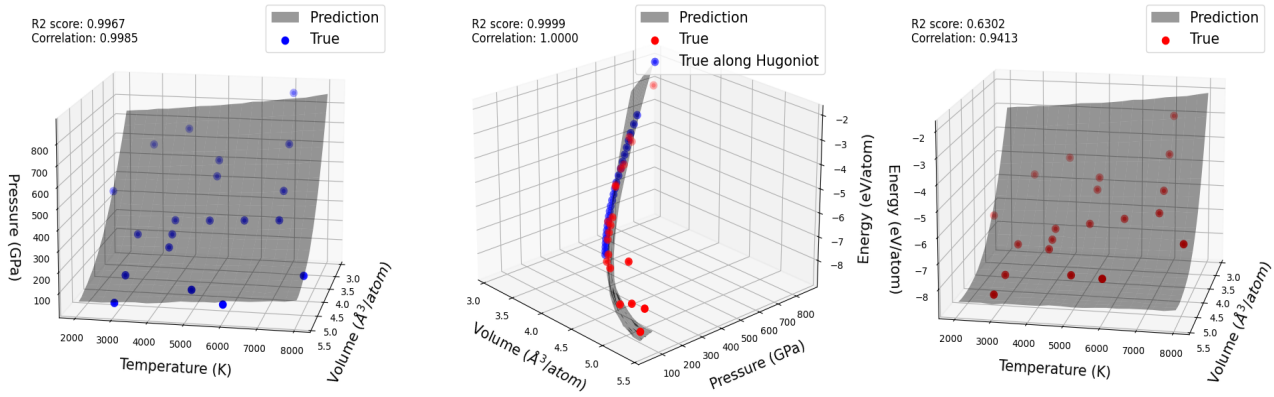

Figure E7. The learned mean surface of $P = P(V,T)$(left), $E = E(V,P)$(middle) and $E = E(V,T)$(right) from the probabilistic model. Same as observed in the deterministic case, the energy information (beyond the sampled ones along the Hugoniot curve, indicated by red dots) were not used in our training, and the model can still produce highly correlated predictions.

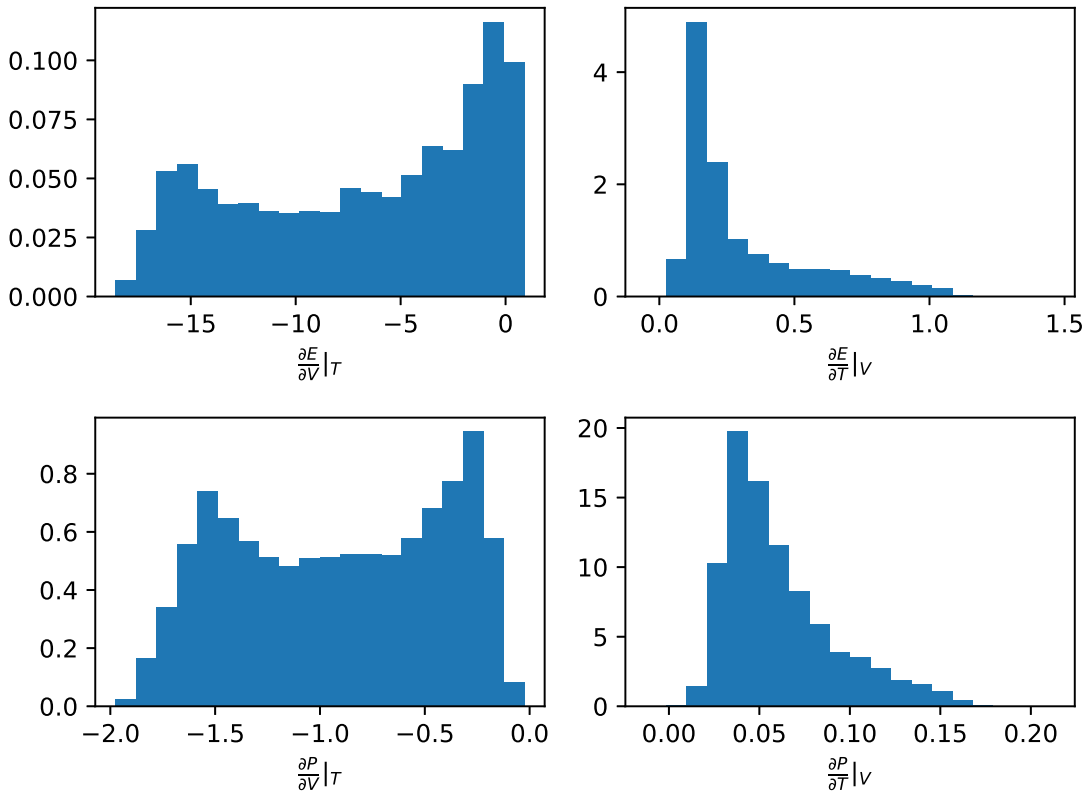

Figure E8. Empirical Histograms of partial derivatives from learned surfaces output by probabilistic model. The sign constraints are also satisfied.

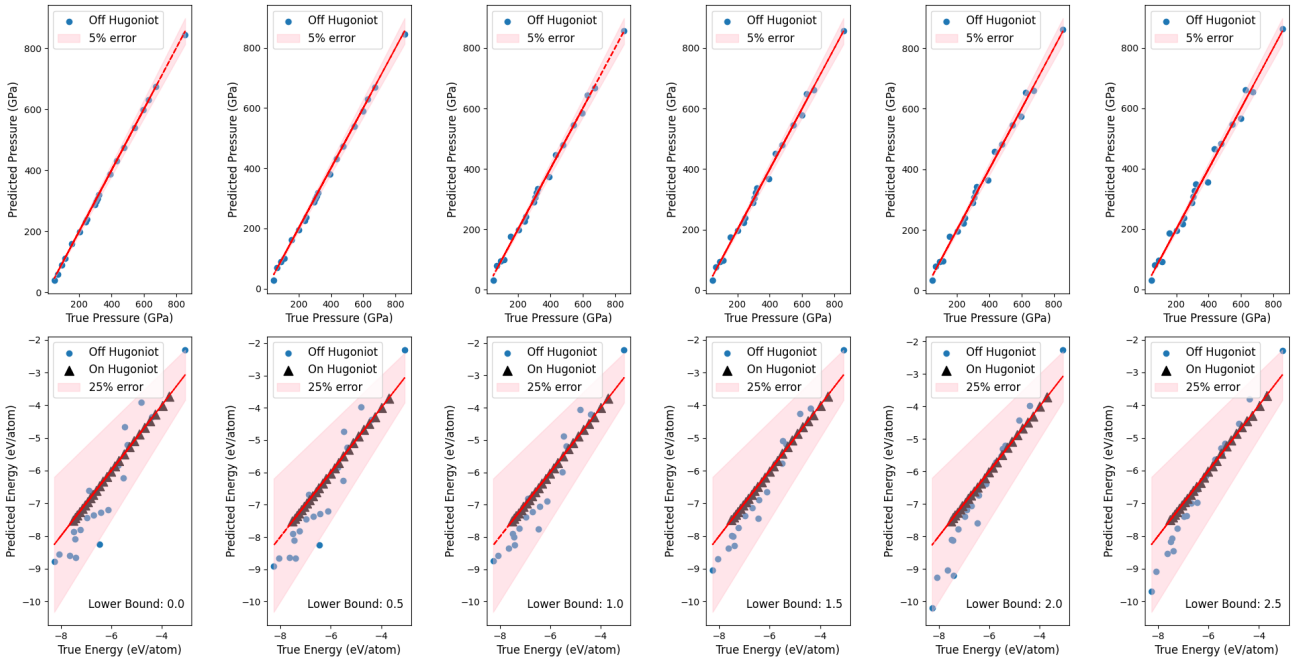

Figure E9. Different prediction results when lower bound of C_V(eV/atom/K/6000) varies for regularization during training.

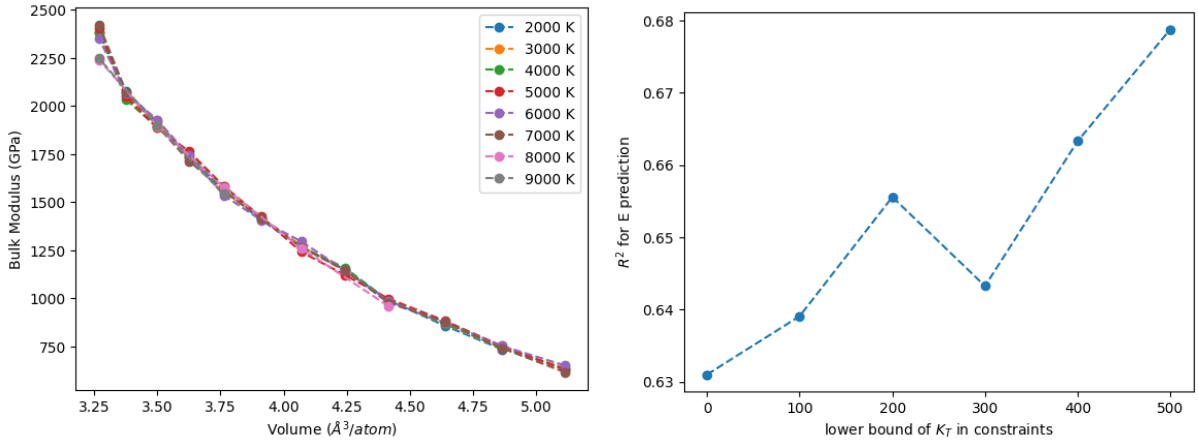

Figure E10. Similar as Section "Case Study - Including more physical prior" in the main manuscript, one can also enforce regularization on the bulk modulus K_T. As an example, we set the lower bound of K_T each time during training at different values: $0, 100, 200, 300, 400, 500$ (GPa) and gather the prediction performance in terms of R^2 score in the prediction of energy off-Hugoniot. Left: True bulk modulus K_T values estimated from data via finite difference scheme on the definition $K_T = -V\frac{dP}{dV}|_T$. Right: R^2 for E prediction when different lower bounds are set in the regularization during training. One can observe from the Figure that the R^2 score gets improved as the lower bound enforced in regularization gets closer to the true lower bound which is around 500 GPa.

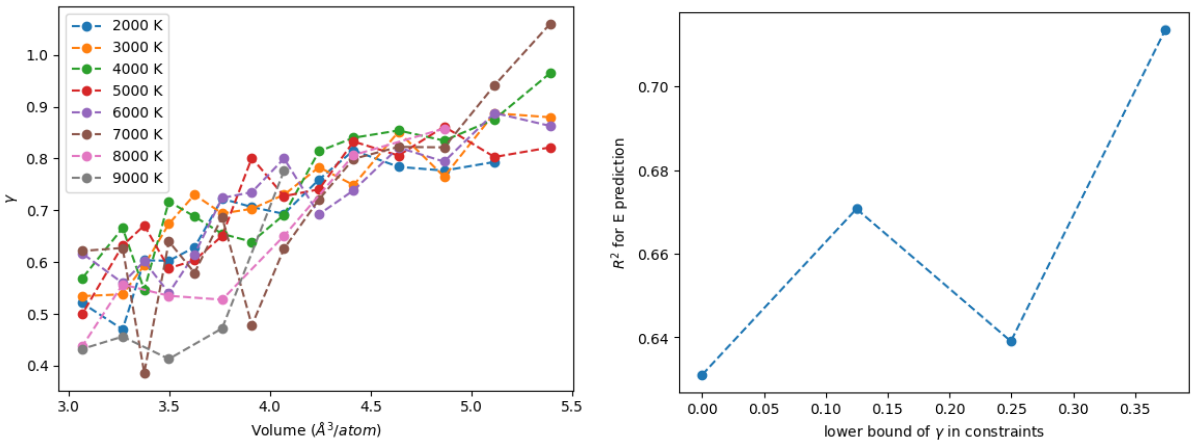

Figure E11. Similar as Section "Case Study - Including more physical prior" in the main manuscript, one can also enforce regularization on the Grüneisen parameter γ. As an example, we set the lower bound of γ each time during training at different values: 0, 0.1248, 0.2497, 0.3745 (*corresponding to values 0, 20, 40, 60 in our codes by multiplying a scalar 1000/6.2415. 0 is the case when no regularization is used*) and gather the prediction performance in terms of R^2 score in the prediction of energy off-Hugoniot. Left: True Grüneisen parameter γ values estimated from data via finite difference scheme on the definition $\gamma = V\frac{dP}{dE}|_V$. Right: R^2 for E prediction when different lower bounds are set in the regularization during training. One can observe from the Figure that the R^2 score gets improved as the lower bound enforced in regularization gets closer to the true lower bound which is around 0.38.

References

1. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
2. Gonzalez-Platas, J., Alvaro, M., Nestola, F. & Angel, R. Eosfit7-gui: a new graphical user interface for equation of state calculations, analyses and teaching. *J. Appl. Crystallogr.* **49**, 1377–1382 (2016).
3. Forbes, J. W. *Shock wave compression of condensed matter: a primer* (Springer, 2012).
4. Sharma, H., Gaffney, J. A., Tsapetis, D. & Shields, M. D. Learning thermodynamically constrained equations of state with uncertainty. *APL Mach. Learn.* **2** (2024).
5. Benedict, L. X. *et al.* Multiphase equation of state for carbon addressing high pressures and temperatures. *Phys. Rev. B* **89**, 224109 (2014).