The Journal of Open Source Software

KIM: Knowledge-Informed Mapping (KIM) Toolkit

Peishi Jiang[1,2], **Aaron Wang**[1], **Susannah M. Burrows**[1], **Naser Mahfouz**[1], **and Xingyuan Chen**[1]

1 Atmospheric, Climate, and Earth Sciences Division, Pacific Northwest National Laboratory, Richland, Washington, USA **2** Civil, Construction and Environmental Engineering, University of Alabama, Tuscaloosa, Alabama, USA

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 21 August 2025
Published: unpublished

Peishi Jiang[1,2], **Aaron Wang**[1], **Susannah M. Burrows**[1], **Naser Mahfouz**[1], **Xingyuan Chen**[1]

[1] Atmospheric, Climate, and Earth Sciences Division, Pacific Northwest National Laboratory, Richland, WA, USA
[2] Civil, Construction and Environmental Engineering, University of Alabama, Tuscaloosa, AL, USA

Summary

We present a Knowledge-Informed Mapping toolkit in Python programming language, named KIM, to optimize the development of the mapping from a vector of inputs $\mathbf{X}$ to a vector of outputs $\mathbf{Y}$. KIM builds on the methodology development of deep learning-based inverse mapping in Jiang et al. (2023) and A. Wang et al. (2025). It involves two key procedures: (1) an exploratory data analysis using information theory to identify the dependency between $\mathbf{X}$ and $\mathbf{Y}$ and filter out both insignificant and redundant inputs through global sensitivity analysis and conditional independence testing; and (2) ensemble learning of using neural networks to account for its structural uncertainty. KIM offers a preliminary understanding of data interdependencies while optimizing the training step with uncertainty accounted for. We demonstrate the applications of KIM through two use cases of developing inverse mappings for learning the parameters of an integrated hydrological model and a large eddy simulation cloud model. We expect this toolkit will be helpful to glue the model data integration for Earth science applications.

Statement of need

Striving for scientific hypothesis testing and discovery, Earth scientists oftentimes develop data-driven mappings – either for inverse modeling, as part of model calibration, or forward modeling, as an emulator. Both approaches benefit from an efficient way of mapping, , that projects from a vector of inputs $\mathbf{X}$ to a vector of outputs $\mathbf{Y}$. While the forward modeling focuses on developmeng an emulator, the inverse modeling involves developing a mapping from model outputs to model parameters, such that once trained, the mapping can directly infer the parameters based on observations. Such mapping approach has seen successes in addressing inverse and forward problems in multiple studies across Earth sciences (Cromwell et al., 2021; HU et al., 2014; Krasnopolsky & Schiller, 2003; Mudunuru et al., 2022).

Nevertheless, constructing the mapping that connects all inputs $\mathbf{X}$ to all outputs $\mathbf{Y}$ is usually challenging due to (1) limited data/simulations for training; (2) uninformative relations between some members of $\mathbf{X}$ and $\mathbf{Y}$; and (3) the structural uncertainty of the mapping . To that, Jiang et al. (2023) and A. Wang et al. (2025) leveraged the idea of integrating scientific knowledge with deep learning (Willard et al., 2022) to develop knowledge-informed mapping (KIM) by using (1) information theory to uncover the dependencies between $\mathbf{X}$ and $\mathbf{Y}$ that guides the design of f and (2) ensemble learning to account for uncertainty due to the model structure error of f. The goal of this paper is to document and open source KIM for a general public usage. Figure 1 shows the general procedures of KIM which are detailed in the next section.

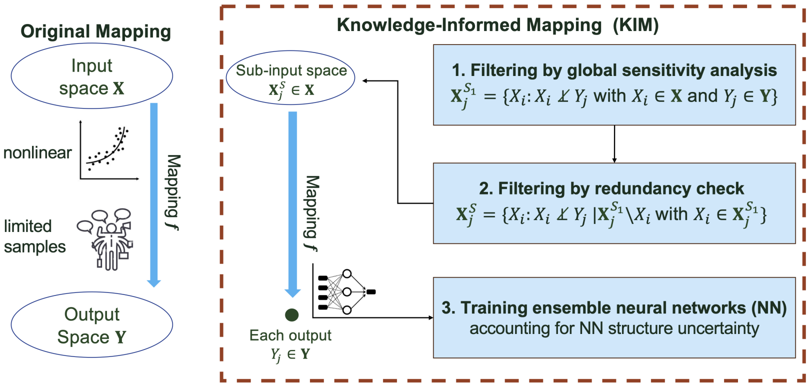
Original Mapping
Input space $\mathbf{X}$
nonlinear
limited samples
Mapping f
Output Space $\mathbf{Y}$
Knowledge-Informed Mapping (KIM)
Sub-input space $\mathbf{X}_j^S \in \mathbf{X}$
Mapping f
Each output $Y_j \in \mathbf{Y}$
1. Filtering by global sensitivity analysis
$\mathbf{X}_j^{S_1} = \{X_i : X_i \not\perp Y_j \text{ with } X_i \in \mathbf{X} \text{ and } Y_j \in \mathbf{Y}\}$
2. Filtering by redundancy check
$\mathbf{X}_j^{S} = \{X_i : X_i \not\perp Y_j \,|\mathbf{X}_j^{S_1}\backslash X_i \text{ with } X_i \in \mathbf{X}_j^{S_1}\}$
3. Training ensemble neural networks (NN)
accounting for NN structure uncertainty

Figure 1: Comparison between KIM and the original mapping.

Mathematical approach

Consider a vector of inputs $\mathbf{X} = [X_1, ..., X_{N_x}]$ and a vector of outputs $\mathbf{Y} = [Y_1, ..., Y_{N_y}]$. The objective is to build up a mapping function f from $\mathbf{X}$ to $\mathbf{Y}$, such that $f : \mathbb{R}^{N_x} \to \mathbb{R}^{N_y}$, based on N_e pairs/realizations of $\mathbf{X}$ and $\mathbf{Y}$. However, it is oftentimes hard to develop an accurate composite f, given the limited training data, partially due to the high computational cost of model ensemble simulation. Instead, it would be desirable to develop a separate inverse mapping f_i for each $Y_j \in \mathbf{Y}$ by using a reduced space $\mathbf{X}_j^S \in \mathbf{X}$ that is most relevant to Y_j, such that $f_j : \mathbb{R}^{N_{xj}} \to \mathbb{R}$ (see examples in Jiang et al. (2023) and A. Wang et al. (2025)). The identification of $\mathbf{X}_j$ for a given Y_j involves a two-step filtering as follows.

Step 1: Filtering by global sensitivity analysis. We first perform a mutual information-based global sensitivity analysis to narrow down a subset $\mathbf{X}_j^{S_1}$, each of which shares zero information with Y_j such that:

$$\mathbf{X}_j^{S_1} = \{X_i : I(X_i; Y_j) \neq 0 \quad \text{with } X_i \in \mathbf{X}\},$$

where $I(X_i; Y_j)$ is the mutual information between X_i and Y_j (Cover & Thomas, 2006). Based on the N_e realizations, I is calculated on the joint probability of X_i and Y_j using either binning method or k-nearest-neighbor method. Following Jiang et al. (2023), a statistical significance test is performed to identify the significant I (i.e., $I(X_i; Y_j) \neq 0$) with a significance level of $1 - \alpha$ on 100 bootstrap samples.

Step 2: Filtering by redundancy check. Then, we conduct a further assessment that filters out any model output in $\mathbf{X}_j^{S_1}$ whose dynamics are redundant to Y_j given the knowledge of other outputs. This is achieved through a conditional independence test using conditional mutual information (Cover & Thomas, 2006) given as:

$$\mathbf{X}_j^S = \{X_i : I(X_i; Y_j | \mathbf{X}_j^{S_1} \backslash X_i) \neq 0 \quad \text{with } X_i \in \mathbf{X}_j^{S_1}\},$$

where $\mathbf{X}_j^{S_1} \backslash X_i$ is the remaining set of $\mathbf{X}_j^{S_1}$ by excluding X_i; $I(X_i; Y_j | \mathbf{X}_j^{S_1} \backslash X_i)$ is the conditional mutual information between X_i and Y_j conditioning on $\mathbf{X}_j^{S_1} \backslash X_i$. $I(X_i; Y_j | \mathbf{X}_j^{S_1} \backslash X_i) = 0$

indicates that X_i and Y_j are independent given the knowledge of $\mathbf{X}_j^{S_1}\backslash X_i$. However, calculating $\mathbf{X}_j^{S_1}\backslash X_i$. $I(X_i;Y_j|\mathbf{X}_j^{S_1}\backslash X_i)$ faces the curse of dimensionality due to the potential high dimension in $\mathbf{X}_j^{S_1}$.

To address this, we leverage the idea of Peter-Clark algorithm for causal inference detection (Spirtes et al., 2001) to evaluate the zeroness of a high-dimensional conditional mutual information by gradually adding conditioning variables. Specifically, we approximated $I(X_i;Y_j|\mathbf{X}_j^{S_1}\backslash X_i)$ via $I(X_i;Y_j|\mathbf{X}_j^{S_2})$, where $\mathbf{X}_j^{S_2}$ is a subset of $\mathbf{X}_j^{S_1}\backslash X_i$ with cardinality ≤ 3. Starting with the cardinality of one, i.e. $|\mathbf{X}_j^{S_2}| = 1$, we conducted statistical significance test on assessing $I(X_i;Y_j|\mathbf{X}_j^{S_2}) = 0$ by exhausting all the combinations out of $\mathbf{X}_j^{S_1}\backslash X_i$ that constitute $\mathbf{X}_j^{S_2}$. We removed X_i from $\mathbf{X}_j^{S}$ when $I(X_i;Y_j|\mathbf{X}_j^{S_2}) = 0$.

Step 3: Uncertainty aware estimation by training ensemble neural networks. For each parameter Y_i, we train an ensemble of fully-connected neural networks by varying the hyperparameters, including the number of hidden layers, the number of hidden neurons, and the learning rate. We split the N_e model realizations into training, validation, and testing dataset. For each model inference, the ensemble learning enables the predictions through weighted mean μ_w and weighted standard deviation σ_w as:

$$\mu_w = \sum_{k=1}^{N_e} w_k \cdot \tilde{y_k}$$
$$\sigma_w = \sqrt{\sum_{k=1}^{N_e} w_k \cdot (\tilde{y_k} - \mu_w)^2},$$

where N_e is the number of ensemble neural networks; $\tilde{y_k}$ is the estimation by the k th neural network; w_k is the weight to the k th prediction and is calculated through the corresponding loss value in the validation dataset $\mathcal{L}_{k,\text{val}}$, such that $w_k = \frac{1/\mathcal{L}_{k,\text{val}}}{\sum_{k=1}^{N_e} 1/\mathcal{L}_{k,\text{val}}}$.

When evaluating the estimation on the test dataset, we further quantified the bias and uncertainty of the prediction as:

$$\text{Bias} = E(|\mu_w - y|)$$
$$\text{Uncertainty} = E(\sigma_w/|y|),$$

where E is the expectation operator and y is the true value.

Examples

We present two applications of KIM in performing inverse modeling, with Jupyter notebook provided in the repository to guide the package usage. For each case, we developed three types of inverse mappings: (1) the original inverse mapping without knowledge-informed, denoted as M_0; (2) the knowledge-informed inverse mapping only using global sensitivity analysis (Step 1), denoted as M_1; and (3) the knowledge-informed inverse mapping using both Step 1 and Step 2, denoted as M_2. 100 neural networks, $N_e = 1$, are trained for each mapping. The remaining configurations can be found in the example jupyter notebook.

Case 1: Calibrating a cloud chamber model. Cloud chamber model has been widely applied as a virtual reality of a true cloud chamber to study turbulence, clouds, and their interactions (Thomas et al., 2019; Aaron Wang, Ovchinnikov, Yang, Schmalfuss, et al., 2024; Aaron Wang, Ovchinnikov, Yang, Cantrell, et al., 2024; Aaron Wang, Krueger, et al., 2024; Aaron Wang et al., 2025). The objective of this example is to estimate two key parameters, i.e., the scaling coefficients of wall fluxes (λ_w) and collision processes (λ_c) using inverse mapping. To that, an ensemble of 513 model runs were generated based on a model set up detailed in A. Wang et al. (2025), by varying the values of the two parameters using Sobol sequence. 27 virtual sensors are configured, each of which 'records' multiple variables including flow properties and cloud properties. The statistics of these variables, calculated over six 5-min periods, are used as the inputs of the inverse mappings, including the temporal standard deviation of vertical velocity, the temporal mean of temperature, the temporal standard deviation of temperature, the temporal mean of supersaturation, the temporal standard deviation of supersaturation, the droplet radius mean, standard deviation, skewness, and kurtosis. Later in Figure 2, these statistics are indicated as Wstd, Tmean, Tstd, SSmean, SSstd, Rmean, Rstd, Rskew, and Rkurt, respectively.

Figure 2 shows the sensitivity analysis and the redundancy check. These two steps greatly reduce the total number of critical model states as the inputs to the inverse mappings. For λ_w, it drops from 1458 first to 1168 (via global sensitivity analysis) and then to 1043 (via redundancy checks). For the logarithm of λ_c, it is reduced to only 336 model states. The corresponding training results in the test dataset are shown in Figure 3. It is obvious that the two KIMs, i.e., M_1 and M_2, outperforms the original inverse mapping that takes all 1043 as the inputs, with reduced bias and uncertainty.

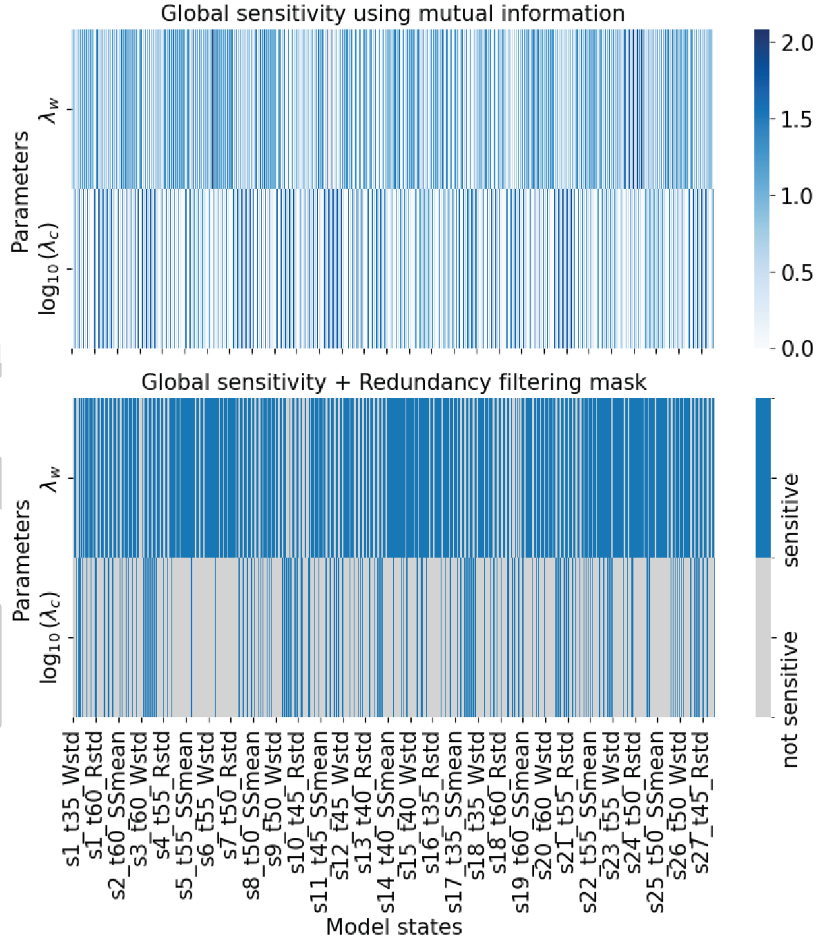

Figure 2: Preliminary analysis of cloud chamber ensemble modeling.

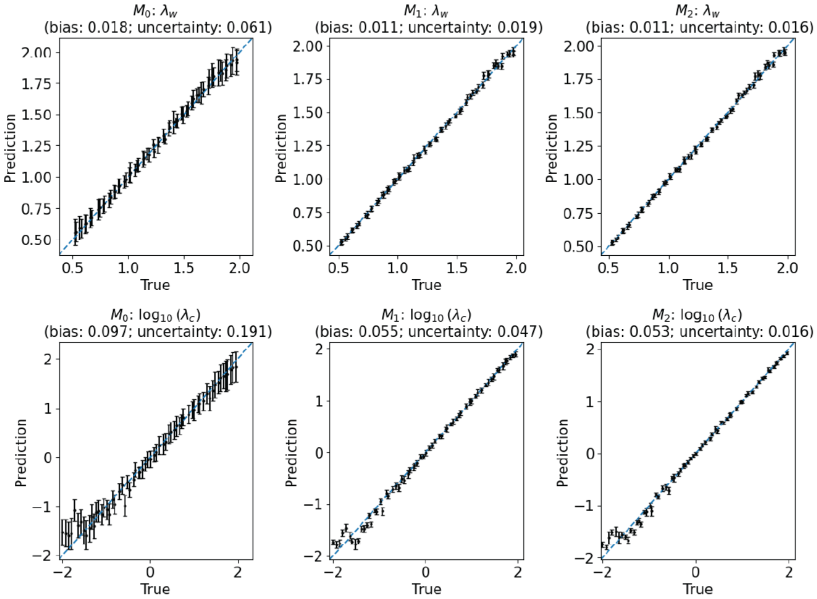

Figure 3: Parameter estimation of the cloud chamber model.

Case 2: Calibrating an integrated hydrological model. The Advanced Terrestrial Simulator (ATS) is an integrated hydrological models used to simulate hydrological fluxes across a watershed (Coon et al., 2019). Here, we calibrated ATS against the streamflow observations at the outlet of Coal Creek watershed, CO, USA. The objective is to estimate eight models parameters categorized into evapotranspiration (ET), snow melting, and subsurface permeability. See Jiang et al. (2023) for more detailed information.

The mutual information-based sensitivity analysis and the redundancy filtering are shown in Figure 4. Similar to the cloud chamber case, the two steps greatly narrow down the streamflows that are useful to inform the estimation of each parameter. Figure 5 shows the corresponding scatter plots between prediction and true for the test dataset, where the bias and uncertainty are drastically lowered by using the two KIMs.

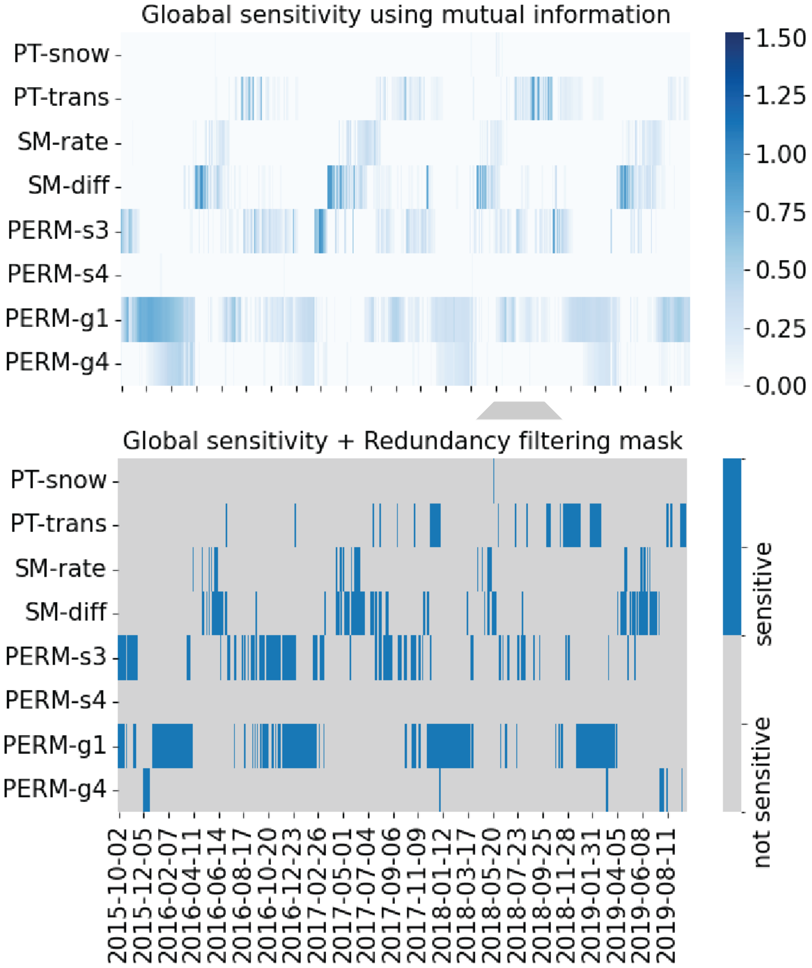

Figure 4: Preliminary analysis of ATS ensemble modeling.

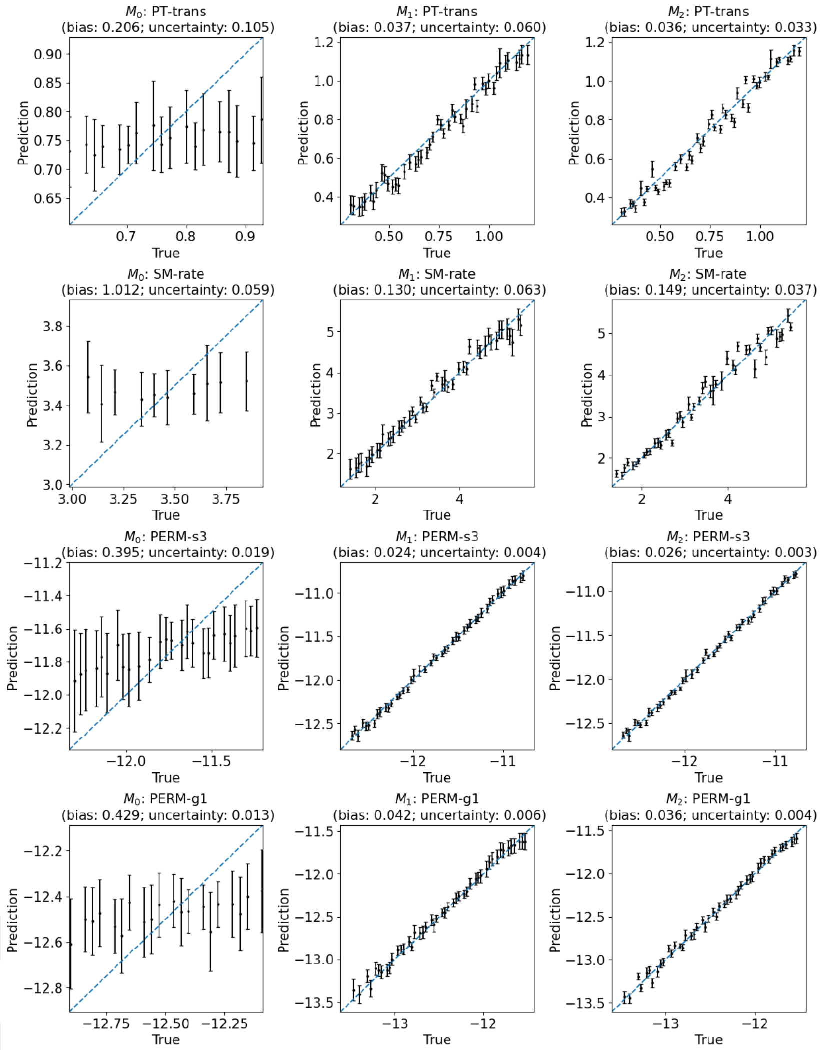

Figure 5: Parameter estimation of the ATS model.

Both examples demosntrates the improved performance of inverse modeling by using KIM. It illustrates the robustness of the proposed method developing data-driven mappings using knowledge-informed techniques and ensemble learning.

Acknowledgements

This work was funded by the Laboratory Directed Research and Development Program at Pacific Northwest National Laboratory. The Laboratory Directed Research and Development Program at Pacific Northwest National Laboratory is a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy. Pacific Northwest National Laboratory is operated for the DOE by Battelle Memorial Institute under contract DE-AC05-76RL01830. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the United States Department of Energy or the United States Government. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paidup, irrevocable, world-wide license to publish, or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

References

Coon, E., Svyatsky, D., Jan, A., Kikinzon, E., Berndt, M., Atchley, A., Harp, D., Manzini, G., Shelef, E., Lipnikov, K., Garimella, R., Xu, C., Moulton, D., Karra, S., Painter, S., Jafarov, E., & Molins, S. (2019). *Advanced terrestrial simulator*. [Computer Software] https://doi.org/10.11578/dc.20190911.1. https://doi.org/10.11578/dc.20190911.1

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory (wiley series in telecommunications and signal processing)*. Wiley-Interscience. ISBN: 0471241954

Cromwell, E., Shuai, P., Jiang, P., Coon, E. T., Painter, S. L., Moulton, J. D., Lin, Y., & Chen, X. (2021). Estimating watershed subsurface permeability from stream discharge data using deep neural networks. *Frontiers in Earth Science*, *9*. https://doi.org/10.3389/feart.2021.613011

HU, Y., YU, X., LI, S., CHEN, G., ZHOU, Y., & GAO, Z. (2014). Improving the accuracy of geological model by using seismic forward and inverse techniques. *Petroleum Exploration and Development*, *41*(2), 208–216. https://doi.org/https://doi.org/10.1016/S1876-3804(14)60024-0

Jiang, P., Shuai, P., Sun, A., Mudunuru, M. K., & Chen, X. (2023). Knowledge-informed deep learning for hydrological model calibration: An application to coal creek watershed in colorado. *Hydrology and Earth System Sciences*, *27*(14), 2621–2644. https://doi.org/10.5194/hess-27-2621-2023

Krasnopolsky, V. M., & Schiller, H. (2003). Some neural network applications in environmental sciences. Part i: Forward and inverse problems in geophysical remote measurements. *Neural Networks*, *16*(3), 321–334. https://doi.org/https://doi.org/10.1016/S0893-6080(03)00027-3

Mudunuru, M. K., Son, K., Jiang, P., Hammond, G., & Chen, X. (2022). Scalable deep learning for watershed model calibration. *Frontiers in Earth Science*, *Volume 10 - 2022*. https://doi.org/10.3389/feart.2022.1026479

Spirtes, P., Glymour, C., & Scheines, R. (2001). *Causation, prediction, and search*. The MIT Press. https://doi.org/10.7551/mitpress/1754.001.0001

Thomas, S., Ovchinnikov, M., Yang, F., Voort, D. van der, Cantrell, W., Krueger, S. K., & Shaw, R. A. (2019). Scaling of an atmospheric model to simulate turbulence and cloud microphysics in the pi chamber. *Journal of Advances in Modeling Earth Systems*, *11*(7), 1981–1994.

Wang, A., Jiang, P., Burrows, S., Glienke, S., Ovchinnikov, M., & Mahfouz, N. (2025). *Inverse mapping of the collision kernel and wall flux scaling in a large-scale convection-cloud chamber using local sensors and knowledge-informed deep learning*.

Wang, Aaron, Krueger, S., Chen, S., Ovchinnikov, M., Cantrell, W., & Shaw, R. A. (2024). Glaciation of mixed-phase clouds: Insights from bulk model and bin-microphysics large-eddy simulation informed by laboratory experiment. *Atmospheric Chemistry and Physics*, *24*(18), 10245–10260.

Wang, Aaron, Ovchinnikov, M., Yang, F., Cantrell, W., Yeom, J., & Shaw, R. A. (2024). The dual nature of entrainment-mixing signatures revealed through large-eddy simulations of a convection-cloud chamber. *Journal of the Atmospheric Sciences*, *81*(12), 2017–2039.

Wang, Aaron, Ovchinnikov, M., Yang, F., Schmalfuss, S., & Shaw, R. A. (2024). Designing a convection-cloud chamber for collision-coalescence using large-eddy simulation with bin microphysics. *Journal of Advances in Modeling Earth Systems*, *16*(1), e2023MS003734.

Wang, Aaron, Schmalfuß, S., Chandrakar, K. K., Kia, H. Z., Yang, F., Ovchinnikov, M., Shaw, R. A., & Choi, Y. (2025). An intercomparison of wall fluxes in a turbulent thermal convection chamber: Direct numerical simulations and wall-modeled large-eddy simulations enhanced by machine learning. *Physics of Fluids*, *37*(4).

Willard, J., Jia, X., Xu, S., Steinbach, M., & Kumar, V. (2022). Integrating scientific knowledge with machine learning for engineering and environmental systems. *ACM Comput. Surv.*, *55*(4). https://doi.org/10.1145/3514228