AI to Learn (AI2L): Human-Centered Guidelines for Black-Box-Free AI and Empirical Law Discovery via Symbolic Regression

Seine A. Shintani*

1) Department of Biomedical Sciences, College of Life and Health Sciences,
Chubu University, Kasugai, Aichi, 487-8501, Japan

2) Center for Mathematical Science and Artificial Intelligence,
Chubu University, Kasugai, Aichi, 487-8501, Japan

3) Institute for Advanced Research, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya-shi, 464-8601, Japan

Abstract

Contemporary generative AI—especially large language models (LLMs)—is rapidly permeating research, education, and healthcare owing to its remarkable efficiency and expressive power. At the same time, AI systems raise serious concerns: their black-box nature, the risk of privacy leakage from input data, the lack of clear accountability for automatically generated outputs, and the substantial energy consumption associated with large-scale models. This paper refines and expands the AI to Learn (AI2L) paradigm: a set of human-centered guidelines that deliberately restrict AI to a learning-support role and require that all final deliverables be free of opaque model components. AI2L rests on four pillars: (1) humans retain sovereignty over final decisions; (2) all components preserved in the deliverable must be explainable and auditable; (3) the risk of information leakage is rigorously minimized; and (4) AI usage is managed from the perspective of energy efficiency and long-term sustainability. Building on recent practical deployments and public outreach activities, we translate these pillars into an operational checklist for everyday AI use, particularly with LLM-based services such as ChatGPT.

The central technical contribution of this paper is an AI2L-style empirical study of symbolic regression. Using synthetic yet physically motivated data sets that obey

*Corresponding author. E-mail: s-shintani@fsc.chubu.ac.jp

Kepler's third law and the Q10 temperature rule, we compare three mainstream black-box regressors (linear regression, cubic polynomial regression, and random forests) with a symbolic regression algorithm. All models achieve high accuracy within the training range; however, only symbolic regression discovers the underlying functional form ($T(r) \propto r^{3/2}$ and $v(T) \propto \exp(\alpha(T - T_{\text{ref}}))$) and extrapolates reliably far beyond the training data. These experiments concretely illustrate how AI2L promotes law-seeking, interpretable modeling rather than mere curve fitting.

We then revisit several real-world applications—Grad-CAM-based visualization of under-polished regions on titanium plates, AI-assisted yet human-owned code generation for classroom seating charts, and a reversible anonymization protocol for safe LLM use in education—and analyze them through the AI2L lens. By acknowledging AI's limitations and hazards while harnessing its strengths, AI2L offers a practical framework for ethical, sustainable, and black-box-free integration of AI into scientific and educational workflows.

1 Introduction

Advances in artificial intelligence technology—most notably the rise of large-scale generative AI—are transforming the ways in which society conducts research, education, and creative work [1, 2]. New applications in natural language processing, image recognition, and programming assistance are emerging almost daily, fundamentally reshaping how knowledge is produced and disseminated [1]. Yet the very convenience that allows users to feed vast amounts of unpublished or sensitive data into these systems conceals two intrinsic dangers: (i) once submitted, the information may effectively reside in the cloud beyond institutional oversight, and (ii) the opaque decision processes of AI models make it difficult for humans to detect misinformation or bias embedded in the outputs [3, 4]. Such risks are unacceptable in high-stakes areas such as medicine and research, where incidents of data leakage and ethical violations have already been reported worldwide [3, 4]. Moreover, training and deploying large models demand enormous computational resources and electricity; recent studies document a steep rise in the carbon footprint of deep learning research as a whole [5, 6].

Rather than advocating the unrealistic option of "living without AI," the present study proposes the AI to Learn (AI2L) paradigm: AI is leveraged as a learning companion under human control, its use is minimized to conserve computational resources, and all traces of black-box components are removed from the final deliverables. Our previous preprint introduced the basic concept of AI2L and described several qualitative case studies. Here, we refine the framework in two ways. First, we translate the four AI2L pillars into a concrete operational checklist that can guide everyday use of LLMs in research and education. Second, we present a set of controlled, AI2L-style experiments on symbolic regression that quantitatively demonstrate the difference between high in-range prediction accuracy and true law discovery. These experiments complement recent work on foundation models that achieve excellent predictive performance without recovering underlying physical laws [11] and illustrate how AI2L can steer AI use toward interpretable and sustainable scientific modeling.

2 Risks and Limitations of Naive use of generative AI

2.1 Data Leakage and Security Concerns

Submitting confidential information directly to an Internet-connected generative AI service—such as ChatGPT—exposes corporate secrets and personal data, previously guarded under strict internal controls, to third-party cloud providers [2]. Indeed, several organizations have already restricted in-house use of such services, and legal debates have arisen under the GDPR and Japan's Act on the Protection of Personal Information [7]. Because model operators may reuse user inputs for continual training or service improvement, the data are irreversibly absorbed into the model's parameter space, making subsequent leakage or misuse difficult to detect or remediate [2, 3]. Persisting in the use of generative AI while ignoring this fundamental risk is indefensible both from an information governance perspective and from the standpoint of broader social responsibility.

2.2 Black-Box Opacity and the Accountability Gap

Most contemporary AI systems, including those based on deep learning, cannot explain why they produce a given output. Techniques such as Grad CAM [8] and SHAP [9] offer partial avenues toward explainable AI (XAI), yet many scholars argue that AI models whose inner workings elude human oversight should not be entrusted with high-stakes decisions [10]. Recent work on foundation models reinforces this caution: Vafa et al. showed that a model able to predict planetary orbits with 99.99% accuracy nevertheless failed to recover the underlying Newtonian laws of motion [11]. In other words, the system mastered pattern recognition but could not extract or generalize the governing principles. Such findings highlight the peril of adopting AI outputs at face value and underscore the need for rigorous human vetting before those outputs are incorporated into any final product.

2.3 Bias, Ethical Pitfalls, and Environmental Burden

Generative AI presents a two-fold challenge: it can reproduce the historical and societal biases embedded in its training data while simultaneously amplifying global energy consumption and environmental impact [4, 5]. Because such systems readily inherit prejudices from their data sets, they may generate discriminatory language or make unfair decisions; in high-stakes domains, opaque outputs of this kind can lead to serious ethical violations [10]. Strubell et al. estimated that training a BERT-scale language model once—augmented with neural architecture search—can emit hundreds of metric tons of CO_2, exacerbating the resource gap between academia and industry [5]. In response, Schwartz et al. introduced the notion of "Green AI," arguing that, given comparable performance, researchers should favor models with superior energy efficiency [6].

AI2L operationalizes this principle by (i) restricting AI usage to the learning-support phase, (ii) distilling the results into lightweight, human-readable code or equations, and (iii) eliminating cloud-based inference whenever possible, thereby slashing power consumption during deployment. For example, the seating-chart workflow described later in this paper relies on Python code generated once by ChatGPT; thereafter the task runs locally with zero GPU resources and no further calls to generative AI services. Thus, AI2L constitutes a practical framework that embodies the ideals of Green AI while maintaining transparency and accountability.

3 The AI to Learn (AI2L) Framework

3.1 Definition and Philosophy

AI to Learn (AI2L) treats artificial intelligence as a temporary assistant that augments human learning and creativity; its guiding rule is that no black-box AI component remains in the final deliverables—whether algorithms, research papers, teaching materials, or production code. Here, removal means that the finished system or knowledge artifact no longer depends on hidden model weights or external cloud APIs: humans must be able to understand and explain every structural element and rationale.

Traditional human-in-the-loop (HIL) design inserts people into the training or inference loop for labeling and feedback, improving performance and adding a safety valve; yet in practice the final decision or execution often rests with the AI system [16]. AI2L diverges fundamentally by limiting AI to a catalyst for thought. The machine rapidly explores vast hypothesis spaces and proposes candidate formulas, designs, or code, but humans perform the truth testing, theory building, and social accountability—ultimately excising the AI black box from the finished work. This approach strengthens accountability while minimizing both data-leakage risk and environmental impact.

3.2 Four Pillars of AI2L

AI2L is anchored in four pillars. For each, we provide both a formal description and a plain-language paraphrase that reflects our outreach activities.

1. **Human sovereignty over final decisions.**
   Formal: The authority and responsibility for all final decisions rest with humans; AI outputs are treated as suggestions or learning materials, never as autonomous actions.
   Plain: AI may help you think, but it never signs the document or makes the call.

2. **Mandatory explainability of all retained components.**
   Formal: Any component that remains in the final deliverable (such as equations, source

code, or decision rules) must be understandable by the human stakeholders; opaque model weights or undocumented APIs are not allowed in high-stakes workflows [10].
Plain: Whatever ends up in the product must be something you could explain on a whiteboard.

3. **Rigorous prevention of information leakage.**
Formal: Inputs to external AI services must be designed under a data-minimization principle; confidential data are either excluded or subjected to reversible anonymization with local control of decryption keys [3, 20].
Plain: Before sending anything to the cloud, assume it may leak, and strip it down until that would no longer be a disaster.

4. **Energy efficiency and sustainability.**
Formal: AI2L adopts the Green AI perspective [6]: large models may be used during the learning-support phase, but routine operations must rely on lightweight implementations that minimize energy and hardware requirements.
Plain: Use big models for thinking once, small models or formulas for running every day.

These pillars operationalize the principles of human oversight, accountability, and sustainability emphasized in the EU AI Act and the NIST AI Risk Management Framework [17]. The fourth pillar is particularly significant: by distilling outputs into lightweight, human-readable programs and avoiding large-model inference at deployment, AI2L offers a concrete, Green-AI-aligned solution that sharply reduces computational resources and CO_2 emissions.

3.3 An Operational AI2L Checklist

To make AI2L usable in everyday practice, we translate the pillars into a short checklist that can be applied to small tasks such as drafting an e-mail, creating a spreadsheet macro, or prototyping a data-analysis pipeline. When using an LLM-based service, we recommend the following five steps:

1. **Write down the objective.**
Example: "Prepare a first draft of an internal FAQ answer" or "Generate boilerplate Python code to reorder rows in a CSV."

2. **Specify the shape of the final deliverable.**
Example: "A human-edited paragraph plus a list of source links" or "A commented script that runs without Internet access."

3. **Prepare the inputs under a no-secrets assumption.**
Remove personal identifiers and confidential values; if necessary, replace them with structurally similar dummy data.

4. **Define how the outputs will be checked.**
   Decide in advance what conditions must be met (e.g., presence of references, bounded number of steps, compliance with internal style guides) and who will inspect the result.

5. **Record cost and lessons learned.**
   For each session, note the approximate time spent, the number of AI queries, and any issues encountered, along with one item to improve next time.

In our teaching practice, we have found that this simple discipline dramatically reduces the temptation to "let the AI do everything" and instead frames it as a learning tool whose contributions remain traceable and auditable.

4 Case Study I: Law-Discovering Symbolic Regression Under AI2L

4.1 Motivation and Setup

Symbolic regression searches for analytic expressions that fit data, combining primitive functions (such as addition, multiplication, or square roots) into candidate formulas [12, 13]. Unlike black-box regressors, symbolic regression can yield human-readable laws that extrapolate beyond the training data when the true relationship lies within the chosen function space.

From an AI2L perspective, symbolic regression exemplifies how AI can serve as a "law-discovery assistant" rather than a pure predictor. To highlight this role, we consider two synthetic but physically motivated laws:

1. Kepler's third law for orbital period as a function of radius.
2. The Q10 temperature rule for reaction rates.

In both cases we generate noisy observations from the known law, train three mainstream regressors (linear regression, cubic polynomial regression, and random forests) and one symbolic regression model, and compare their performance inside and far outside the training range. All code is written in Python using open-source libraries; the exact parameters are reported below so that the experiments can be reproduced without any hidden components.

4.2 Kepler's Third Law: Predicting Orbital Periods

4.2.1 Data generation

Under a circular-orbit approximation, Kepler's third law implies that the orbital period T scales with the radius r as

$$T_{\text{true}}(r) = kr^{3/2}, \tag{1}$$

where $k > 0$ is a constant. Without loss of generality we set $k = 1$. We then draw 40 radii uniformly from the range $0.5 \leq r \leq 2.0$ and add Gaussian noise with standard deviation $\sigma = 0.05$ to obtain observed periods:

$$T_{\text{obs}}(r_i) = T_{\text{true}}(r_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma^2). \tag{2}$$

Figure 1 shows that the noisy observations closely follow the underlying $r^{3/2}$ curve.

For evaluation we consider two sets of test radii: an *interpolation range* $0.5 \leq r \leq 2.0$ (densely sampled) and an *extrapolation range* $2.0 \leq r \leq 10.0$.

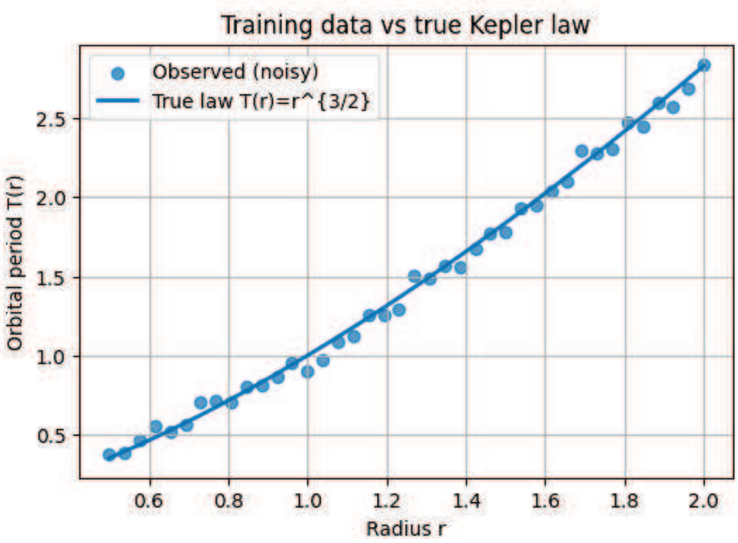


Figure 1: Synthetic training data for Kepler's third law. Blue points: noisy observations T_{obs}. Solid line: true law $T_{\text{true}}(r) = r^{3/2}$.

4.2.2 Baseline regressors

We first train three standard regressors on the training set $\{(r_i, T_{\text{obs}}(r_i))\}$:

- **Linear regression** on the scalar feature r.
- **Cubic polynomial regression**, implemented as linear regression on polynomial features $(1, r, r^2, r^3)$.

- **Random forest regression** with 200 trees and maximum depth 3, using r as the sole feature.

For each model we compute the root-mean-square error (RMSE) in the interpolation and extrapolation ranges with respect to the true law $T_{\text{true}}(r)$. Table 1 summarizes the results.

Table 1: RMSE of baseline and symbolic-regression models for Kepler's law. The training range is $0.5 \le r \le 2.0$; extrapolation is evaluated on $2.0 \le r \le 10.0$.

Model	RMSE (interpolation)	RMSE (extrapolation)
Linear regression	0.060	7.87
Polynomial (degree 3)	0.020	43.5
Random forest	0.055	15.4
Symbolic regression	1.3×10^{-16}	1.3×10^{-15}

Figure 2 visualizes the baseline predictions. All three models track the true curve well within the training range, but their behavior diverges dramatically outside it. Linear regression grows too slowly; the cubic polynomial eventually bends downward and becomes negative; the random forest saturates near a constant value.

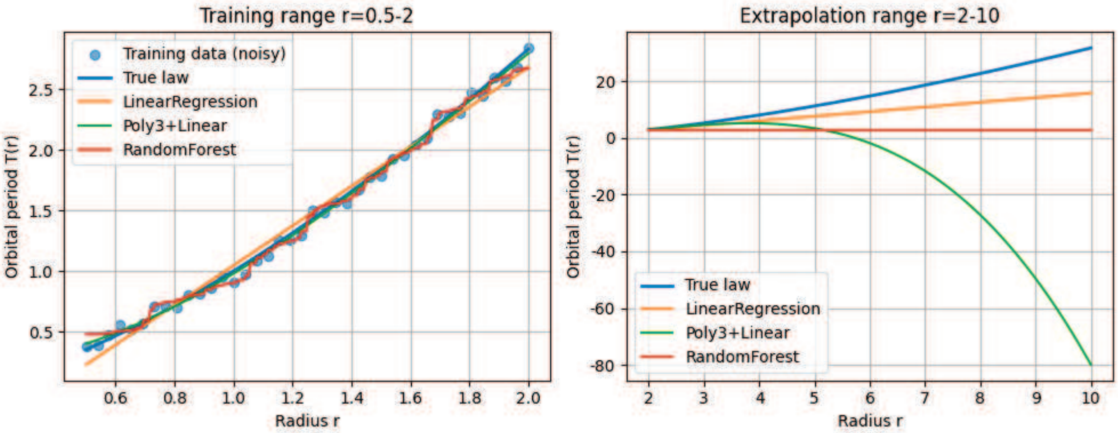


Figure 2: Baseline regressors for Kepler's law. Left: interpolation range $0.5 \le r \le 2.0$; all models approximate the true curve. Right: extrapolation range $2.0 \le r \le 10.0$; none of the black-box models reproduces the correct growth.

4.2.3 Symbolic regression

We next apply a symbolic regression algorithm (gplearn's `SymbolicRegressor`) with function set $\{+, -, \times, \div, \sqrt{\ }\}$. The algorithm starts from random expressions of r and iteratively evolves them to minimize squared error on the training data, with a parsimony penalty to discourage unnecessarily complex formulas.

The best expression found after 25 generations is

$$T_{\mathrm{SR}}(r) = r\sqrt{r}. \tag{3}$$

This is exactly the Kepler law $r^{3/2}$. Consequently, the interpolation and extrapolation RMSE values are at numerical round-off level (Table 1). Figure 3 shows that the symbolic-regression curve coincides with the true law over the entire domain, while the baseline methods deviate strongly in the extrapolation regime.

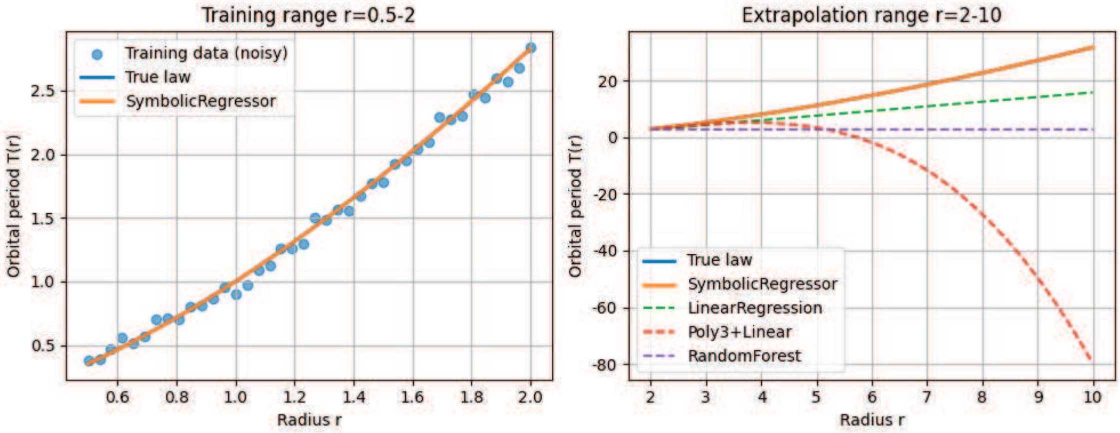


Figure 3: Symbolic regression for Kepler's law. Left: interpolation range; the symbolic-regression curve overlaps the true law. Right: extrapolation range; only symbolic regression continues to match the true law, while baseline models (dashed lines) deviate.

From an AI2L perspective, this experiment is exemplary: the true data-generating law is explicitly documented, the algorithm's function set is simple and transparent, and the final result is a closed-form expression that any student can interpret and reuse without relying on a trained model object.

4.3 Q10 Temperature Rule: Exponential Dependence of Reaction Rates

4.3.1 Data generation

The Q10 rule is an empirical law that describes how reaction rates increase with temperature. Assuming a reference temperature $T_{\mathrm{ref}} = 20°\mathrm{C}$ and Q10 coefficient $Q_{10} = 2$, we define

$$v_{\mathrm{true}}(T) = \exp\left(\alpha(T - T_{\mathrm{ref}})\right), \qquad \alpha = \frac{\ln Q_{10}}{10}, \tag{4}$$

so that raising the temperature by 10°C doubles the rate.

We generate 50 temperatures uniformly in the interval $0°\mathrm{C} \leq T \leq 40°\mathrm{C}$, compute $v_{\mathrm{true}}(T)$, and add Gaussian noise with standard deviation 0.01 to obtain observed rates $v_{\mathrm{obs}}(T)$. Figure 4 shows the resulting data and true curve. Extrapolation is evaluated on 40°C $\leq T \leq$ 80°C.

For numerical stability we use a centered feature

$$x = \frac{T - T_{\text{ref}}}{20}, \tag{5}$$

which is fed to all models.

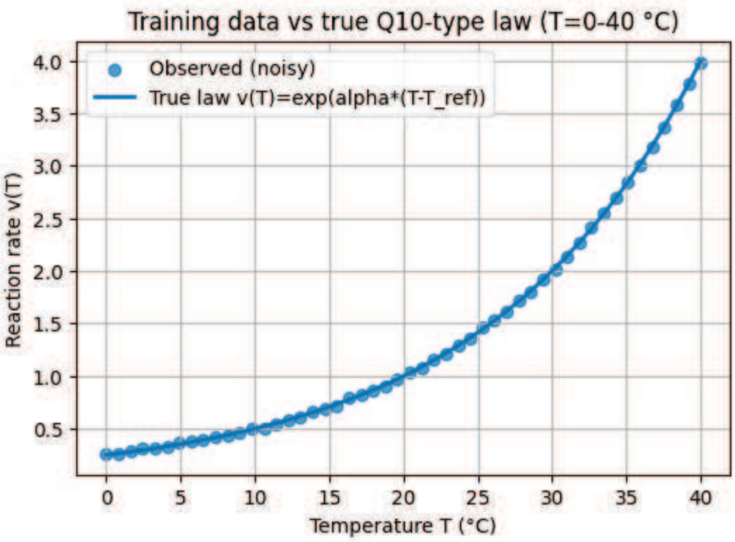


Figure 4: Synthetic training data for the Q10 temperature rule. Blue points: noisy observations $v_{\text{obs}}(T)$. Solid line: true law $v_{\text{true}}(T) = \exp\{\alpha(T - T_{\text{ref}})\}$.

4.3.2 Baseline regressors

We reuse the same three baseline models as in the Kepler experiment, now trained on the feature x. Table 2 reports the RMSE values.

Table 2: RMSE of baseline and symbolic-regression models for the Q10 rule. The training range is 0°C $\leq T \leq$ 40°C; extrapolation is evaluated on 40°C $\leq T \leq$ 80°C.

Model	RMSE (interpolation)	RMSE (extrapolation)
Linear regression	0.341	23.0
Polynomial (degree 3)	0.0138	12.7
Random forest	0.0233	24.2
Symbolic regression	8.8×10^{-4}	5.1×10^{-2}

Within the training range, polynomial regression and random forests achieve low error, whereas linear regression underfits. However, all three black-box models diverge from the true exponential growth in the extrapolation range (Figure 5).

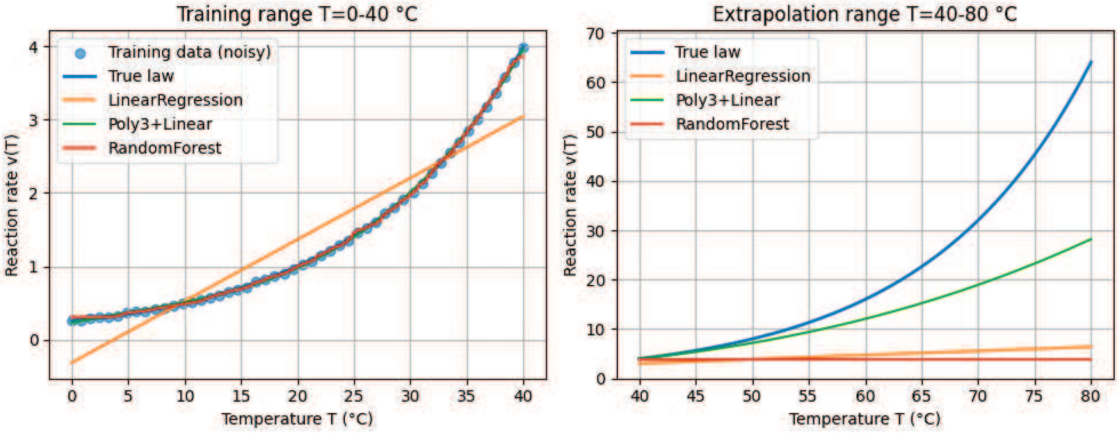


Figure 5: Baseline regressors for the Q10 rule. Left: interpolation range 0°C $\leq T \leq$ 40°C; polynomial regression and random forests fit well. Right: extrapolation range 40°C $\leq T \leq$ 80°C; none of the models reproduces the steep exponential increase.

4.3.3 Symbolic regression

To encourage discovery of an exponential law we configure the symbolic regressor with function set $\{+, \times, \exp\}$. The best expression found is

$$v_{\mathrm{SR}}(T) = \exp\Big(x + 0.386x\Big), \qquad x = \frac{T - T_{\mathrm{ref}}}{20}. \tag{6}$$

Rewriting, we obtain

$$v_{\mathrm{SR}}(T) = \exp\left(\beta(T - T_{\mathrm{ref}})\right), \qquad \beta \approx \frac{1 + 0.386}{20} \approx 0.0693, \tag{7}$$

which corresponds to $Q_{10} \approx e^{10\beta} \approx 2.00$. Thus the symbolic regressor has effectively rediscovered the Q10 rule. Its RMSE is an order of magnitude smaller than that of the polynomial and random-forest baselines, both inside and outside the training range (Table 2 and Figure 6).

4.4 Discussion: Symbolic Regression as an AI2L Tool

These two synthetic experiments illustrate several key points.

First, high interpolation accuracy does not guarantee correct extrapolation or law discovery. Black-box models such as random forests can nearly interpolate the training data while failing dramatically outside that range, especially when the true relationship involves unbounded growth.

Second, symbolic regression can act as an AI2L-compatible law-discovery engine. By specifying simple, physically motivated function sets, we guide the search toward interpretable

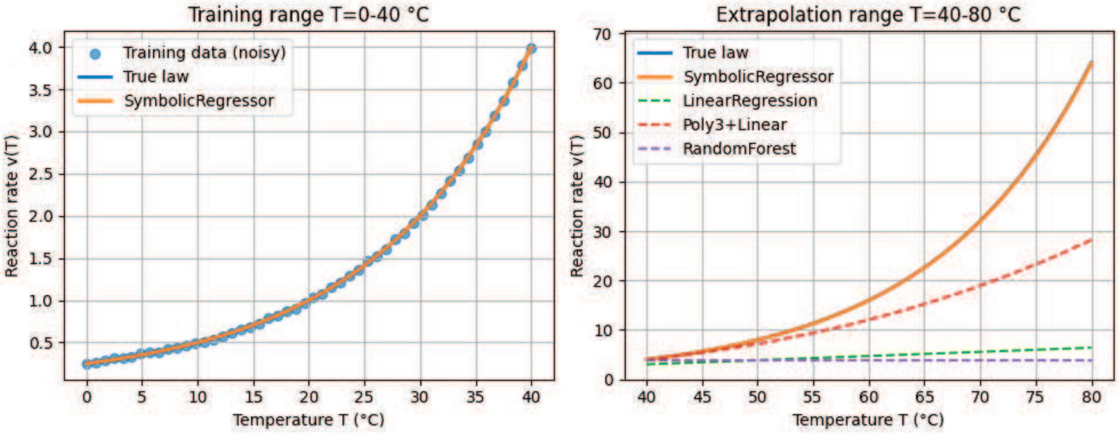


Figure 6: Symbolic regression for the Q10 rule. Left: interpolation range; symbolic regression matches the true exponential curve. Right: extrapolation range; only symbolic regression continues to follow the exponential growth, whereas baseline models (dashed lines) underestimate it.

expressions. The final outputs are compact formulas that can be printed in a textbook, derived on a whiteboard, or implemented in a few lines of code. No trained model object or cloud API is required at deployment.

Third, the entire workflow is transparent and privacy preserving. All data are synthetically generated from documented equations; there is no opportunity for personal or confidential information to leak. Every modeling choice—from noise level to model hyperparameters—is recorded, enabling precise reproducibility.

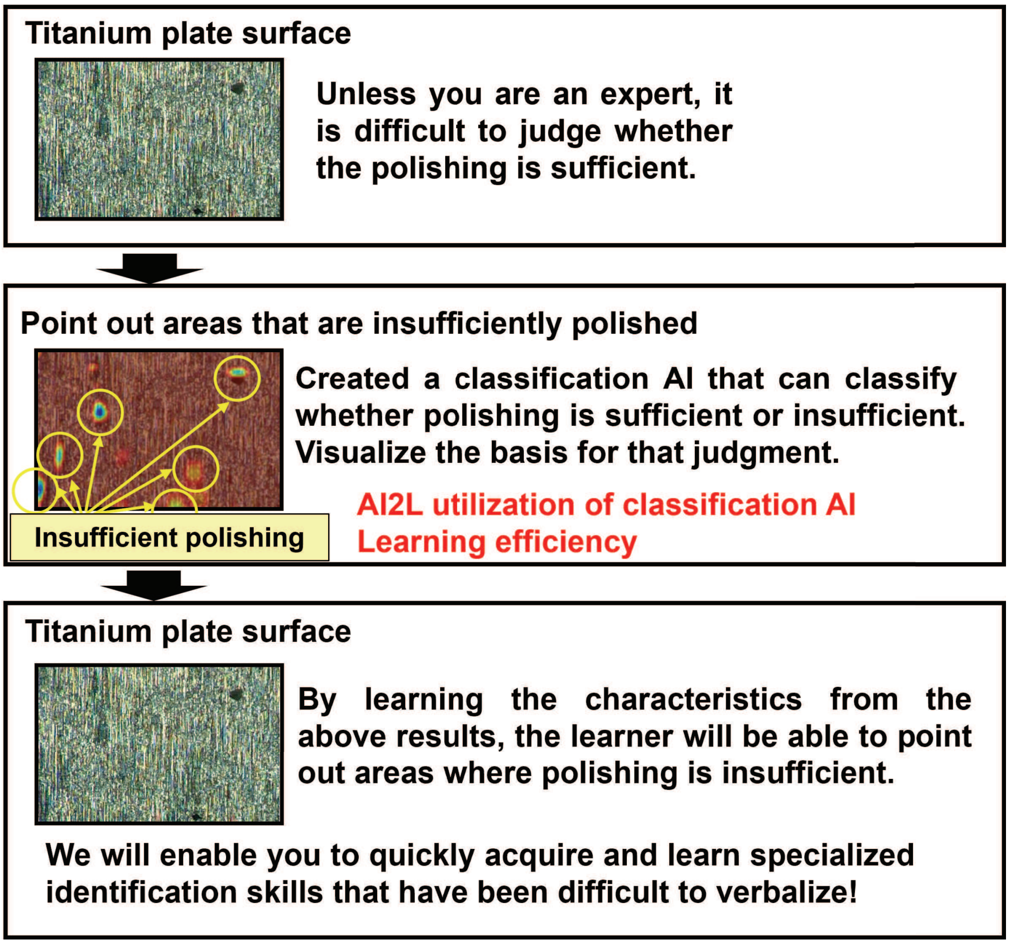
These characteristics align perfectly with the four AI2L pillars. Extending the same methodology to real experimental data, as in our previous work on sarcomeric oscillations and chaordic homeodynamics [18, 19, 14, 15], allows AI to accelerate hypothesis generation while leaving theory building and responsibility squarely in human hands.

5 Case Study II: Visualizing Tacit Knowledge in Titanium-Plate Inspection

We next summarize an application in which AI2L helps externalize tacit human expertise.

We built a convolutional neural network (CNN) to classify the surface condition of titanium plates—adequately versus inadequately polished—and used Grad CAM to visualize the evidence for each decision (Figure 7) [8]. The heat map overlays consistently highlighted microscopic dark regions of residual titanium oxide in images labeled “under polished,” precisely the areas that experienced inspectors had been focusing on tacitly. This enabled us to codify a clear quality-control rule—“presence of dark TiO_2 residue → insufficient polishing”—

and to develop a short training module through which non-experts could master the skill in minutes.



Titanium plate surface

Unless you are an expert, it is difficult to judge whether the polishing is sufficient.

Point out areas that are insufficiently polished

Created a classification AI that can classify whether polishing is sufficient or insufficient. Visualize the basis for that judgment.

AI2L utilization of classification AI
Learning efficiency

Titanium plate surface

By learning the characteristics from the above results, the learner will be able to point out areas where polishing is insufficient.

We will enable you to quickly acquire and learn specialized identification skills that have been difficult to verbalize!

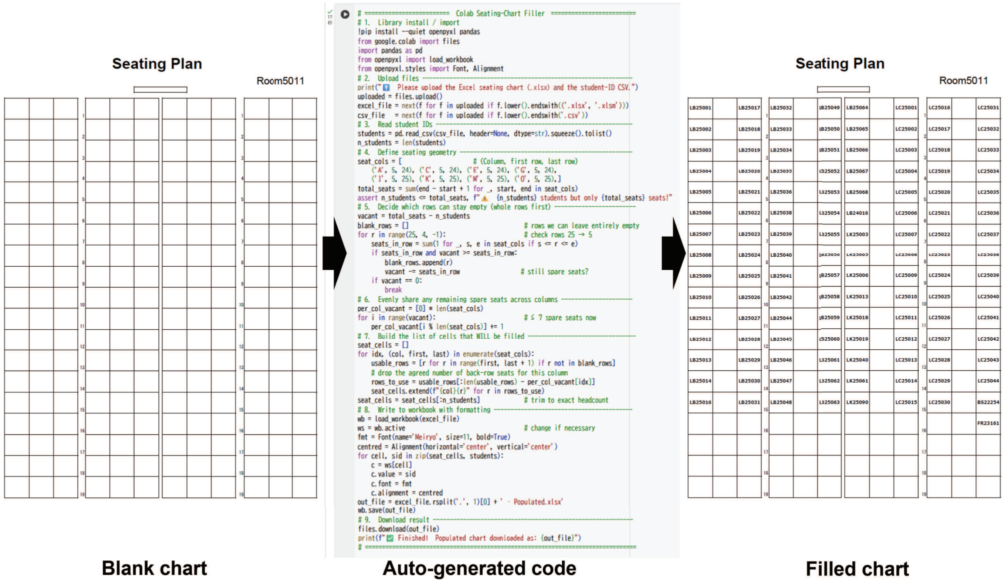
Figure 7: Grad-CAM visualization of the titanium-polish classifier. Top: CNN prediction alongside the Grad-CAM heat map; the yellow circle marks the dark oxide region that triggered the "under polished" label. Bottom: when used as instructional material, the visualization allows learners to grasp in a short time the expert cue—dark regions corresponding to residual TiO_2—that was previously difficult to articulate.

From the AI2L standpoint, the AI system handled only two tasks during the learning-support phase: (i) training the polishing-quality classifier and (ii) externalizing tacit knowledge via Grad CAM. Routine inspection was then shifted to a manual procedure in which humans directly look for the highlighted dark regions; the AI model was removed from the operational loop, eliminating black-box dependence. What remains on the shop floor is the human understanding—"black oxide spots signal under polishing"—and a rapid manual check, while the AI served merely as a short-term accelerator.

6 Case Study III: Human-Owned AI-Generated Code and Safe LLM Use

6.1 A Seating-Chart Workflow Fully Aligned with AI2L

The seemingly mundane task of assigning more than 120 student IDs to an eight-column seating chart typically requires at least thirty minutes and is prone to copy-and-paste errors. Applying AI2L principles, we compressed the entire workflow to 17 seconds without uploading any confidential data to a generative AI service. The process comprised four stages (Figure 8).



```python
# ========================  Colab Seating-Chart Filler  ========================
# 1.  Library install / import
!pip install --quiet openpyxl pandas
from google.colab import files
import pandas as pd
from openpyxl import load_workbook
from openpyxl.styles import Font, Alignment
# 2.  Upload files ------------------------------------------------------------
print("⬆ Please upload the Excel seating chart (.xlsx) and the student-ID CSV.")
uploaded = files.upload()
excel_file = next(f for f in uploaded if f.lower().endswith(('.xlsx', '.xlsm')))
csv_file   = next(f for f in uploaded if f.lower().endswith('.csv'))
# 3.  Read student IDs --------------------------------------------------------
students = pd.read_csv(csv_file, header=None, dtype=str).squeeze().tolist()
n_students = len(students)
# 4.  Define seating geometry -------------------------------------------------
seat_cols = [                         # (Column, first row, last row)
    ('A', 5, 24), ('C', 5, 24), ('E', 5, 24), ('G', 5, 24),
    ('I', 5, 25), ('K', 5, 25), ('M', 5, 25), ('O', 5, 25),]
total_seats = sum(end - start + 1 for _, start, end in seat_cols)
assert n_students <= total_seats, f"⚠ {n_students} students but only {total_seats} seats!"
# 5.  Decide which rows can stay empty (whole rows first) ---------------------
vacant = total_seats - n_students
blank_rows = []                                  # rows we can leave entirely empty
for r in range(25, 4, -1):                       # check rows 25 → 5
    seats_in_row = sum(1 for _, s, e in seat_cols if s <= r <= e)
    if seats_in_row and vacant >= seats_in_row:
        blank_rows.append(r)
        vacant -= seats_in_row                  # still spare seats?
    if vacant == 0:
        break
# 6.  Evenly share any remaining spare seats across columns ------------------
per_col_vacant = [0] * len(seat_cols)
for i in range(vacant):                          # ≤ 7 spare seats now
    per_col_vacant[i % len(seat_cols)] += 1
# 7.  Build the list of cells that WILL be filled -----------------------------
seat_cells = []
for idx, (col, first, last) in enumerate(seat_cols):
    usable_rows = [r for r in range(first, last + 1) if r not in blank_rows]
    # drop the agreed number of back-row seats for this column
    rows_to_use = usable_rows[:len(usable_rows) - per_col_vacant[idx]]
    seat_cells.extend(f"{col}{r}" for r in rows_to_use)
seat_cells = seat_cells[:n_students]             # trim to exact headcount
# 8.  Write to workbook with formatting ---------------------------------------
wb = load_workbook(excel_file)
ws = wb.active                                   # change if necessary
fmt = Font(name='Meiryo', size=11, bold=True)
centred = Alignment(horizontal='center', vertical='center')
for cell, sid in zip(seat_cells, students):
    c = ws[cell]
    c.value = sid
    c.font = fmt
    c.alignment = centred
out_file = excel_file.rsplit('.', 1)[0] + ' - Populated.xlsx'
wb.save(out_file)
# 9.  Download result ---------------------------------------------------------
files.download(out_file)
print(f"✅ Finished!  Populated chart downloaded as: {out_file}")
# ============================================================================
```

LB25001		LB25017	1	LB25032		LB25049	LB25064		LC25001	1	LC25016		LC25031
LB25002		LB25018	2	LB25033		LB25050	LB25065		LC25002	2	LC25017		LC25032
LB25003		LB25019	3	LB25034		LB25051	LB25066		LC25003	3	LC25018		LC25033
LB25004		LB25020	4	LB25035		LB25052	LB25067		LC25004	4	LC25019		LC25034
LB25005		LB25021	5	LB25036		LB25053	LB25068		LC25005	5	LC25020		LC25035
LB25006		LB25022	6	LB25038		LB25054	LB24016		LC25006	6	LC25021		LC25036
LB25007		LB25023	7	LB25039		LB25055	LK25003		LC25007	7	LC25022		LC25037
LB25008		LB25024	8	LB25040		LB25056	LK25005		LC25008	8	LC25023		LC25038
LB25009		LB25025	9	LB25041		LB25057	LK25006		LC25009	9	LC25024		LC25039
LB25010		LB25026	10	LB25042		LB25058	LK25013		LC25010	10	LC25025		LC25040
LB25011		LB25027	11	LB25044		LB25059	LK25018		LC25011	11	LC25026		LC25041
LB25012		LB25028	12	LB25045		LB25060	LK25019		LC25012	12	LC25027		LC25042
LB25013		LB25029	13	LB25046		LB25061	LK25040		LC25013	13	LC25028		LC25043
LB25014		LB25030	14	LB25047		LB25062	LK25061		LC25014	14	LC25029		LC25044
LB25016		LB25031	15	LB25048		LB25063	LK25090		LC25015	15	LC25030		BS22254
			16							16			FR23161
			17							17			
			18							18			
			19							19			

Figure 8: AI-assisted yet human-owned seating-chart workflow. Left: blank Excel template (Room 5011). Centre: excerpt of the Python script auto-generated by ChatGPT from a single prompt. Right: completed chart produced in 17 s; the algorithm leaves the rear rows empty by evenly distributing surplus seats.

1. **One-shot code generation.** A single English prompt was submitted to ChatGPT (o3 pro). Within seconds the model produced an open-source Python script that (i) reads a headerless CSV of student IDs, (ii) maps the designated Excel seat ranges, (iii) evenly distributes surplus seats to the back rows, and (iv) formats every entry in Meiryo 11 pt, bold, and center aligned.

2. **Human review and hardening.** The script was downloaded, variable names were refactored, and explicit `assert` statements plus exception handling were added. This audit erased the last vestige of black-box dependence, turning the code into a fully transparent asset.

3. **Execution in a controlled sandbox.** For convenience, the hardened script was executed on Google Colab, but only the blank Excel template and an anonymized ID

list—information acceptable for external storage—were uploaded. No generative-AI inference occurs at runtime; Colab functions solely as a commodity Python interpreter, so the model weights of ChatGPT are never exposed to the data and associated privacy risks are eliminated [3].

4. **Download and local reuse.** The populated seating chart was downloaded and archived. The identical script can subsequently be run on an on-premise PC—without internet connectivity or GPU acceleration—incurring negligible energy cost and zero future dependence on any large model [5].

This workflow epitomizes the balanced ethos of AI2L: a generative model is leveraged once to accelerate development, after which day-to-day operations proceed independently of large-scale AI. The outcome is a lightweight, auditable, and easily modifiable script that fosters user autonomy while minimizing cloud exposure and carbon footprint [6].

6.2 Reconciling Data Anonymisation with Generative-AI Use

Many educational and research tasks do not require uploading real data at all. If the inputs supplied to a generative AI service are replaced with structurally similar dummy data, model prototyping and code generation proceed perfectly well. The seating-chart script described above, for instance, was fully validated with a CSV containing random IDs; no confidential file ever left the local environment.

When genuine data must be processed via AI—for example, grading student essays with highly variable formatting—AI2L introduces reversible anonymization. Identifiers such as student numbers and names are replaced en bloc with either Fernet-based symmetric encryption or random tokens, ensuring that the file transmitted to the external service contains nothing that a human reader could trace back to an individual. The encryption key and mapping table remain in an offline lab computer; scores or feedback returned by the AI are decrypted locally. In this way, automated assessment is achieved without compromising personal privacy.

Before any upload the following checks are mandatory:

1. Encrypt all direct identifiers.
2. Generalize quasi-identifiers (e.g., replace “Department of Bioengineering, Year 3” with “STEM, upper division”).
3. Scan free-text fields for residual re-identification risks and redact as necessary.

Only when residual risk falls below the operational threshold—and a human explicitly confirms this—may the data be sent.

In summary, AI2L prescribes a three-layer guardrail:

1. Avoid attachments altogether whenever possible.
2. Substitute dummy data for development and testing.
3. When real data are unavoidable, apply reversible anonymization and keep decryption strictly local.

This protocol allows practitioners to benefit from generative AI while preventing data leakage and simultaneously cultivating higher data-governance literacy.

7 Theoretical and Societal Significance

7.1 True Understanding versus Pattern Recognition

The impressive predictive power of large models is often misconstrued as proof that "AI understands the world." Yet both our synthetic experiments and Vafa et al.'s inductive bias probe [11] show that a system able to predict orbital positions with 99.99% accuracy can still fail to recover Newtonian mechanics. High-accuracy prediction and deep understanding are not the same; bridging that gap requires human critical thinking and theory building [10].

AI2L assigns AI the role of scanning the vast hypothesis space, while humans verify, interpret, and generalize the results into universal laws. In this scheme, pattern recognition—the "Keplerian" level of knowledge—serves as a springboard for humans to leap toward explanatory principles—the "Newtonian" level. Our symbolic-regression case studies provide a concrete, accessible demonstration of this transition.

7.2 A Unified Demand for AI Ethics, Governance, and Energy Efficiency

International frameworks such as the EU AI Act and the NIST AI RMF designate human oversight, accountability, and sustainability as indispensable [17]. In practice, however, operational models that simultaneously suppress (a) black-box dependence, (b) personal-data leakage, and (c) massive power consumption are rare. AI2L offers a three-in-one protocol:

1. **Explainability**: all AI outputs are distilled into human-readable formulas or code; no external service remains in the production pipeline [10].
2. **Information protection**: dummy data and reversible anonymization keep re-identification risk to a minimum [3, 20].

3. **Green AI implementation**: generative models are used only during the learning-support phase, while routine tasks run locally in lightweight form [5, 6].

By meeting these three goals at once, AI2L uniquely realizes an AI-specific triple bottom line, integrating ethical, legal, and environmental imperatives into a single, practical methodology.

8 Challenges and Future Directions for AI2L Adoption

Because AI2L deliberately limits AI autonomy and assigns ultimate responsibility to humans, it imposes short-term costs—such as (i) the human effort required for code audits and formula verification, and (ii) the relinquishment of some automated pipeline features offered by AI APIs. Yet these costs yield long-term returns in the form of quality assurance, risk reduction, and enhanced researcher skill sets. Key practical challenges and prospects include:

1. **Feedback loops from real-world practice**. Domain-specific artifacts—such as anonymization templates for medical records or grading rubrics for education—should be shared in open repositories so that AI2L workflows can evolve through continuous community feedback.
2. **Standardizing evaluation metrics**. Quantitative benchmarks are needed: XAI scores based on Grad CAM or SHAP [8, 9], energy indicators such as watt hours per inference, and other measures that allow the "degree of AI2L compliance" to be assessed objectively.
3. **Horizontal expansion to other sectors**. While this paper focused on materials inspection, biophysics, and educational administration, the benefits of AI2L are even greater in fields where human accountability is paramount—government document review, legal support, automotive maintenance, and more.
4. **Integration into policy and education**. University curricula and corporate training programs should incorporate "AI2L practicum" modules, enabling researchers and engineers to master AI while managing its limits in a systematic way.

In building a sustainable AI society, AI2L offers a healthy counterbalance to the myth of "fully autonomous AI." By uniting black-box elimination, accountability, data protection, and energy conservation, the framework is poised to become a natural default for near-future AI governance and education.

Acknowledgements

All experiments and manuscript writing were solely performed by the author. This work was supported by JSPS KAKENHI Grant Number JP25K00269 (Grant-in-Aid for Scientific

Research (C), Project Title: “Elucidation of Myosin Molecular Dynamics Associated with Sarcomere Morphological Changes in the Intracellular Environment”).

References

[1] R. Bommasani, D. A. Hudson, E. Adeli, *et al.*, “On the Opportunities and Risks of Foundation Models,” *arXiv preprint* arXiv:2108.07258, 2021.

[2] OpenAI, “GPT-4 Technical Report,” *arXiv preprint* arXiv:2303.08774, 2023.

[3] N. Carlini, F. Tramèr, E. Wallace, *et al.*, “Extracting Training Data from Large Language Models,” in *Proc. 30th USENIX Security Symp.*, 2021, pp. 2633–2650.

[4] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, “On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?” in *Proc. ACM Conf. Fairness, Accountability, and Transparency (FAccT)*, 2021, pp. 610–623.

[5] E. Strubell, A. Ganesh, and A. McCallum, “Energy and Policy Considerations for Deep Learning in NLP,” in *Proc. 57th Annu. Meeting Assoc. Comput. Ling. (ACL)*, 2019, pp. 3645–3650.

[6] R. Schwartz, J. Dodge, N. Smith, and O. Etzioni, “Green AI,” *Communications of the ACM*, vol. 63, no. 12, pp. 54–63, 2020.

[7] T. Kuru, “Lawfulness of the Mass Processing of Publicly Accessible Online Data to Train Large Language Models,” *International Data Privacy Law*, Oct. 2024.

[8] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626.

[9] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017, pp. 4765–4774.

[10] C. Rudin, “Stop Explaining Black Box Machine Learning Models for High-Stakes Decisions and Use Interpretable Models Instead,” *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[11] K. Vafa, P. G. Chang, A. Rambachan, and S. Mullainathan, “What Has a Foundation Model Found? Using Inductive Bias to Probe for World Models,” in *Proc. 42nd Int. Conf. Machine Learning (ICML)*, PMLR 267, pp. 60727–60747, 2025. See also arXiv:2507.06952.

[12] M. Schmidt and H. Lipson, “Distilling Free-Form Natural Laws from Experimental Data,” *Science*, vol. 324, no. 5923, pp. 81–85, Apr. 2009.

[13] S.-M. Udrescu and M. Tegmark, "AI Feynman: A Physics-Inspired Method for Symbolic Regression," *Science Advances*, vol. 6, no. 16, eaay2631, Apr. 2020.

[14] S. A. Shintani, "Hyperthermal Sarcomeric Oscillations Generated in Warmed Cardiomyocytes Control Amplitudes with Chaotic Properties While Keeping Cycles Constant," *Biochem. Biophys. Res. Commun.*, vol. 611, pp. 8–13, 2022.

[15] S. A. Shintani, "Chaordic Homeodynamics: The Periodic Chaos Phenomenon Observed at the Sarcomere Level and its Physiological Significance," *Biochem. Biophys. Res. Commun.*, vol. 760, 151712, 2025.

[16] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza, "Power to the People: The Role of Humans in Interactive Machine Learning," *AI Magazine*, vol. 35, no. 4, pp. 105–120, 2014.

[17] E. Tabassi, "Artificial Intelligence Risk Management Framework (AI RMF) 1.0," National Institute of Standards and Technology, Gaithersburg, MD, USA, NIST AI 100-1, Jan. 2023.

[18] S. A. Shintani, K. Oyama, N. Fukuda, and S. Ishiwata, "High-Frequency Sarcomeric Auto-Oscillations Induced by Heating in Living Neonatal Cardiomyocytes of the Rat," *Biochem. Biophys. Res. Commun.*, vol. 457, no. 2, pp. 165–170, 2015.

[19] S. A. Shintani, T. Washio, and H. Higuchi, "Mechanism of Contraction Rhythm Homeostasis for Hyperthermal Sarcomeric Oscillations of Neonatal Cardiomyocytes," *Scientific Reports*, vol. 10, 20468, 2020.

[20] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," *Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.