nature machine intelligence

Article

https://doi.org/10.1038/s42256-024-00964-x

Causal chambers as a real-world physical testbed for AI methodology

Received: 17 April 2024

Accepted: 26 November 2024

Published online: 15 January 2025

Check for updates

Juan L. Gamella ✉, Jonas Peters & Peter Bühlmann

In some fields of artificial intelligence, machine learning and statistics, the validation of new methods and algorithms is often hindered by the scarcity of suitable real-world datasets. Researchers must often turn to simulated data, which yields limited information about the applicability of the proposed methods to real problems. As a step forward, we have constructed two devices that allow us to quickly and inexpensively produce large datasets from non-trivial but well-understood physical systems. The devices, which we call causal chambers, are computer-controlled laboratories that allow us to manipulate and measure an array of variables from these physical systems, providing a rich testbed for algorithms from a variety of fields. We illustrate potential applications through a series of case studies in fields such as causal discovery, out-of-distribution generalization, change point detection, independent component analysis and symbolic regression. For applications to causal inference, the chambers allow us to carefully perform interventions. We also provide and empirically validate a causal model of each chamber, which can be used as ground truth for different tasks. The hardware and software are made open source, and the datasets are publicly available at causalchamber.org or through the Python package causalchamber.

Methodological research in artificial intelligence (AI), machine learning and statistics often develops without a concrete application in mind. Many impactful advances in these fields have been made in this way, and there are important theoretical questions that are studied outside the context of a particular application. Crucially, progress also relies on having access to high-quality, real-world datasets, which benefits methodological and theoretical researchers by helping them steer research in meaningful directions, relaxing assumptions that are unlikely to hold in practice, as well as developing methodologies that may work well on a variety of real-world problems.

However, for some research areas, particularly nascent ones, it can be difficult to find real-world datasets that provide a ground truth suitable to validate new methods and check foundational assumptions that underlie theoretical work. This is because new fields come with new requirements in terms of ground truth, and few or no datasets may have been collected that already satisfy them. For example, for most sub-fields of causal inference[1–3], we require data from phenomena in which the underlying causal relationships are already exquisitely understood or for which carefully designed intervention experiments are available. For symbolic regression[4,5], the data must follow a known, closed-form mathematical expression, for example, a natural law in a controlled experimental environment. For the different types of representation learning[6,7], we may need data for which there are some latent 'generating factors' that we can measure directly. Such datasets can be difficult to obtain in practice, and few exist for these tasks. As a result, researchers are often limited to synthetic data produced by computer simulations, which may fall short of answering how well a particular method works in practice.

This is where we believe our work can contribute. We have constructed two physical devices that allow the inexpensive and automated collection of data from two well-understood, real physical systems (Fig. 1). The devices, which we call causal chambers, consist of a light tunnel and a wind tunnel (Fig. 2). They are, in essence, computer-controlled laboratories to manipulate and measure different variables of the physical system they contain.

Seminar for Statistics, ETH Zurich, Zurich, Switzerland. ✉e-mail: juangamella@gmail.com

https://doi.org/10.1038/s42256-024-00964-x

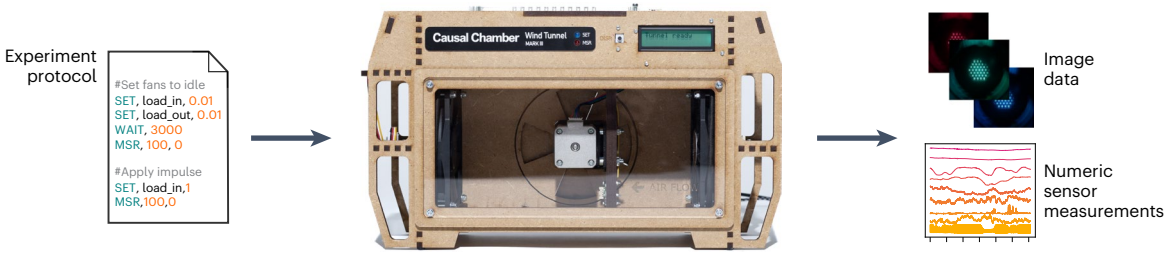


Fig. 1 | Data collection workflow. The user provides an experiment protocol consisting of step-by-step instructions describing the data collection procedure, which the chamber then carries out without human supervision. The instructions specify when and to which values the actuators and sensor parameters should be set. They also specify when the measurements of all the variables should be taken and at which frequency, at a maximum of 10 Hz for the light tunnel and 7 Hz for the wind tunnel. Actuators and sensor parameters can also be set automatically by the chamber as a function of other variables in the system, such as sensor measurements. This allows introducing additional complexity for some validation tasks, as described in the 'A testbed for algorithms' section.

We believe that the chambers are well suited to substantially improve the validation of methodological advancements across machine learning and statistics, by providing real datasets with a ground truth for fields in which such datasets are otherwise scarce or non-existent. This is accomplished through two key properties of the chambers. First, the underlying physical systems are well understood, in the sense that the relationships between most variables are described by first principles and natural laws involving linear, nonlinear and differential equations (Supplementary Sections III and IV provide a detailed description with carefully designed experiments). This allows us to provide ground truths for various tasks, including a causal model of each chamber. Second, we can manipulate the systems in a controlled and automated way, quickly producing vast amounts of data. Furthermore, the chambers produce data of different modalities, including independent and identically distributed, time-series and image data, allowing us to provide validation tasks for a wide range of methodologies.

To illustrate the practical use of the chambers, we perform case studies in causal discovery, out-of-distribution generalization, change point detection, independent component analysis (ICA) and symbolic regression (see the 'Case studies' section and Figs. 5 and 6). Our choice constitutes only an initial selection, and we believe many other possibilities exist.

Our work complements existing datasets from more complex real-world systems for which a ground truth is not or only partially available[8], as well as efforts to produce synthetic data that mimics such systems[5,9–14]. Although good performance on the chambers is not guaranteed to carry over to more complex systems, we believe that the chambers can serve as a sanity check for foundational assumptions and algorithms that are intended to work in a variety of settings.

A list of all the datasets we currently provide can be found at https://causalchamber.org, together with a description of the experimental procedures used to collect them. To allow other researchers to build their own chambers, blueprints, component lists and source code are available in ref. 15.

The causal chambers

Each chamber is a machine that contains a simple physical system and allows us to measure and manipulate some of its variables. The chambers contain a variety of sensors, for example, to measure light intensity or barometric pressure. To manipulate the physical system, actuators allow us to control, for example, the brightness of a light source or the speed at which fans turn. Each sensor can also be manipulated by modifying some of its parameters, such as the oversampling rate or reference voltage.

Throughout this paper, we refer to the actuators and sensor parameters as the manipulable variables of the chamber. A programmable onboard computer controls all the sensor parameters and actuators, enabling the chambers to conduct experiments and collect data without human supervision (Fig. 1). As a result, the chambers can quickly produce vast amounts of data, up to millions of observations or tens of thousands of images per day.

In the remainder of this section, we give an overview of each chamber, its physical system and some of the measured variables. Figure 2 provides diagrams of the chambers and their main components, and a detailed description of all the variables can be found in Supplementary Section II.

The wind tunnel

The wind tunnel (Fig. 2a,c) is a chamber with two controllable fans that push air through it and barometers that measure air pressure at different locations. A hatch at the back of the chamber controls an additional opening to the outside. A microphone measures the noise level of the fans, and a speaker allows for an independent effect on its reading.

The tunnel provides data from 32 numerical and categorical variables (Fig. 4a shows some examples), of which 11 are sensor measurements and 21 correspond to actuators and sensor parameters that can be manipulated. For example, we can control the load of the two fans ($L_{\text{in}}, L_{\text{out}}$) and measure their speed ($\tilde{\omega}_{\text{in}}, \tilde{\omega}_{\text{out}}$), the current they draw ($\tilde{C}_{\text{in}}, \tilde{C}_{\text{out}}$) and the resulting air pressure inside the chamber ($\tilde{P}_{\text{dw}}, \tilde{P}_{\text{up}}$) or at its intake ($\tilde{P}_{\text{int}}$). We can manipulate the sensor parameters like the oversampling rate of the barometers ($O_{\text{dw}}, O_{\text{up}}, O_{\text{int}}, O_{\text{amb}}$) or the resolution of the speed sensors ($T_{\text{in}}, T_{\text{out}}$), further affecting their measurements. In the circuit that drives the speaker, we can manipulate the potentiometers (A_1, A_2) that control the amplification, monitoring the resulting signal at different points of the circuit ($\tilde{S}_1, \tilde{S}_2$) and through the microphone output ($\tilde{M}$).

The light tunnel

The light tunnel (Fig. 2b,d) is a chamber with a controllable light source at one end and two linear polarizers mounted on rotating frames. The relative angle between the polarizers dictates how much light passes through them (Fig. 4c,e) and sensors measure the light intensity before, between and after the polarizers. A camera on the side opposite the light source allows taking images from inside the tunnel.

The tunnel provides image data (Fig. 4e) and 41 numerical and categorical variables (Fig. 4b–d), out of which 32 can be manipulated. For example, we can control the intensity of the light source at three different wavelengths (R, G, B) and measure the drawn electric current ($\tilde{C}$). Using motors, we can rotate the polarizer frames to the desired angles (θ_1, θ_2) and measure the effect on light intensity at different wavelengths ($\tilde{I}_1, \tilde{I}_2, \tilde{I}_3, \tilde{V}_1, \tilde{V}_2, \tilde{V}_3$). We can manipulate sensor parameters like the exposure time of the camera (T_{lm}) or the photodiode used by the light sensors (D^I_1, D^I_2, D^I_3), further affecting the readings of these sensors.

A testbed for algorithms

The chambers are designed to provide a testbed for a variety of algorithms from AI, machine learning and statistics. To set up validation

https://doi.org/10.1038/s42256-024-00964-x

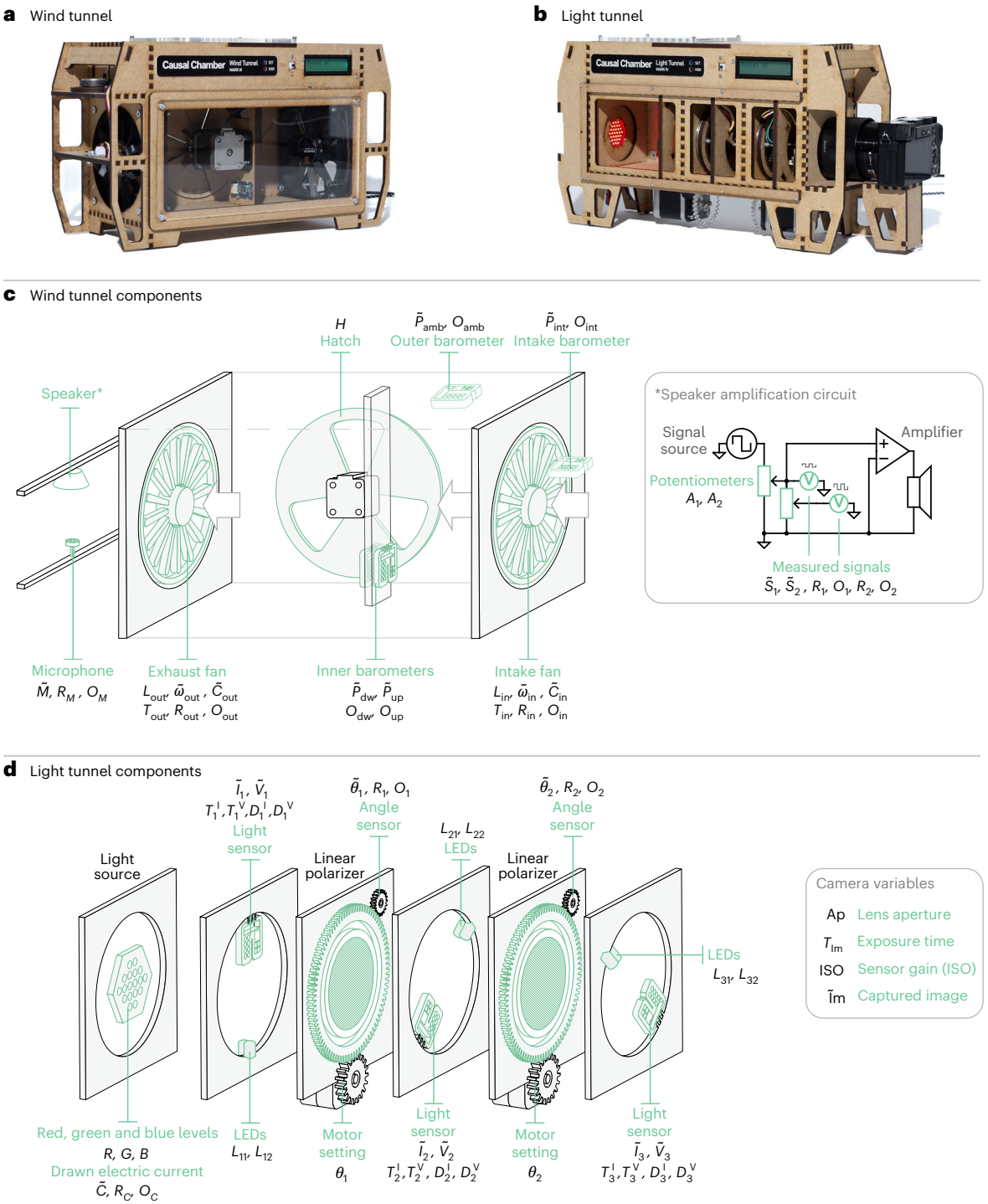


Fig. 2 | The causal chambers. a, The wind tunnel. **b**, The light tunnel with the front panel removed to show its inner components. **c**,**d**, Diagrams of the chambers and their main components, including the amplification circuit that drives the speaker of the wind tunnel and the variables for the light tunnel camera. The variables measured by the chambers are also displayed. The sensor measurements are denoted by a tilde '~'. Manipulable variables, namely, actuators and sensor parameters, are shown in bold symbols (shown as non-bold text elsewhere in the text). A detailed description of each variable is given in Supplementary Section II.

tasks, we rely on two key properties of the chambers: that the encapsulated physical system is well understood and that we can manipulate it. For example, by manipulating actuators, we can evaluate a learned causal model in its prediction of interventional distributions. By contrast, when the relationships between actuators and sensors are well described by a natural law, we can set up a symbolic regression task in which we try to recover it from data. These are some examples of the tasks we set up in the 'Case studies' section, but many other possibilities exist.

In Fig. 3, we provide a graphical representation of the physical system in each chamber under different configurations, in the form of a directed graph relating its variables. In their most basic form, the chambers operate in the standard configuration, where the values of all the actuators and sensor parameters are explicitly given by the user

https://doi.org/10.1038/s42256-024-00964-x

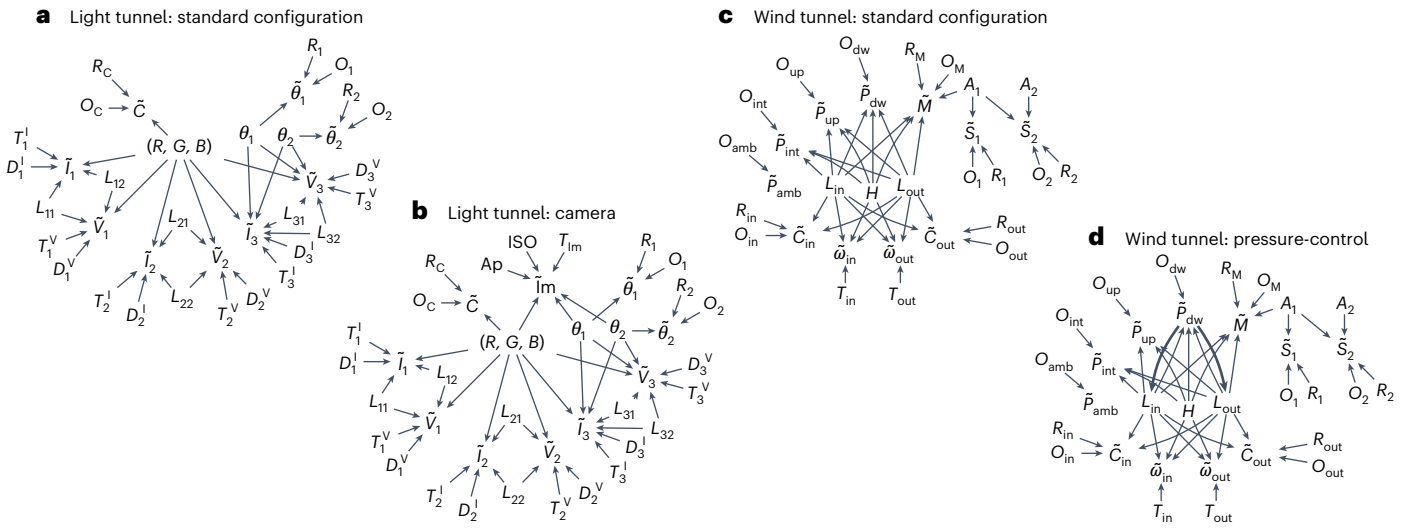


Fig. 3 | Representation of the known effects for different chamber configurations. The bold symbols correspond to manipulable variables, such as actuators and sensor parameters (shown as non-bold text elsewhere in the text). Sensor measurements are denoted by a tilde. **a**,**c**, Standard configurations of the chambers. **b**, Camera configuration of the light tunnel, including images from the light tunnel ($\tilde{\mathbf{I}}$ m) and the camera parameters (Ap, ISO, T_{Im}). **d**, Pressure control configuration of the wind tunnel, where the load fans L_{in}, L_{out} are set by a control mechanism to maintain the chamber pressure $\tilde{P}_{\mathrm{dw}}$ at a given level. Each effect (edge in the graph) is described in detail with the targeted experiments in Supplementary Section III.

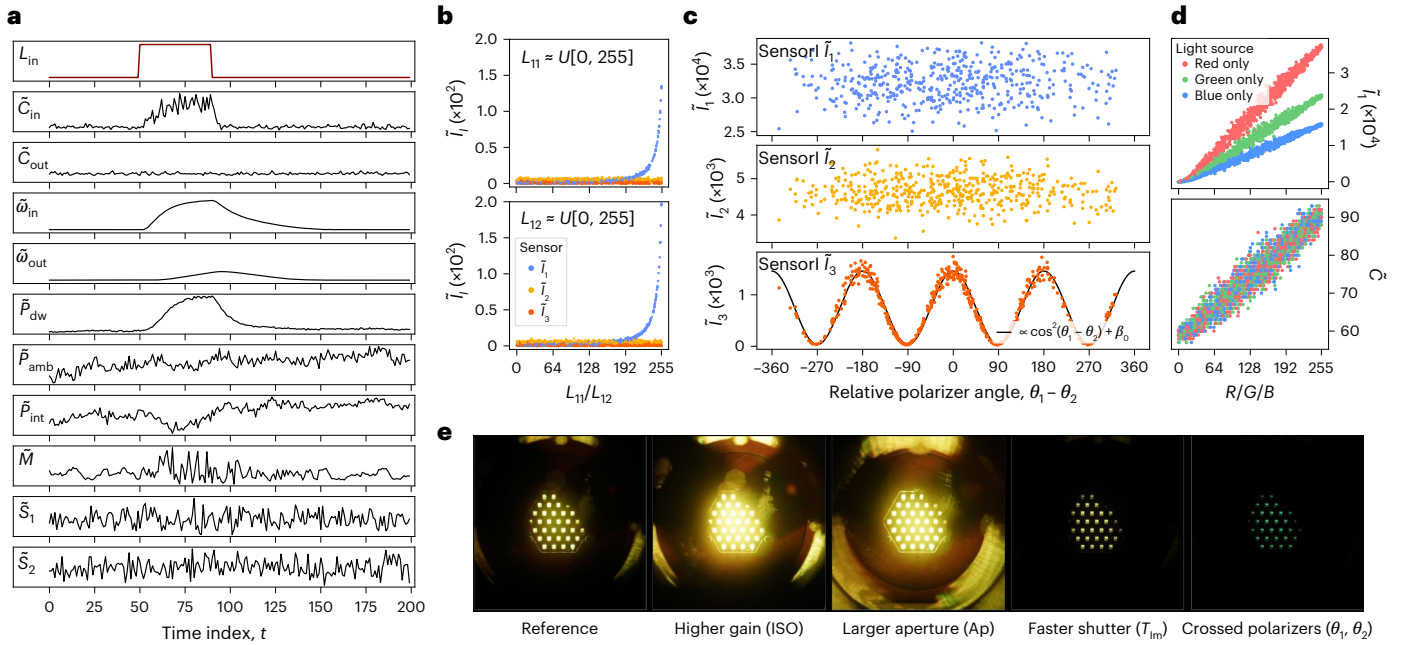


Fig. 4 | Examples of data produced by the chambers. **a**, Numeric time-series data produced by the wind tunnel under an impulse on the intake fan load (L_{in}, red), affecting other variables in the system. **b**,**c**, Numerical data from the light tunnel illustrating the effect of LED brightness (L_{11}, L_{12}) and polarizer angles (θ_1, θ_2) on the light-intensity readings ($\tilde{I}_1$, $\tilde{I}_2$, $\tilde{I}_3$). **d**, Effect of the light-source setting (R, G, B) on the light-intensity reading of the first sensor ($\tilde{I}_1$) and drawn current ($\tilde{C}$). **e**, Examples of images from the light tunnel for a fixed light-source setting (reference) and interventions on other variables that affect the resulting image.

in the experiment protocol (Fig. 1). The light tunnel operates without the camera to allow for the fastest measurement rate.

For additional flexibility in setting up the validation tasks, the chambers can also operate in extended configurations. For example, these can include additional variables, such as those from the camera in the light tunnel (Fig. 3b) or additional sensors included in the future. Furthermore, the extended configurations also allow us to assign the value of actuators and sensor parameters as a function of other variables in the system, such as sensor measurements. The assignment is done automatically by the computer onboard the chamber and allows us to introduce additional complexity into the system. For example, the pressure control configuration of the wind tunnel (Fig. 3d) implements a control mechanism that continuously updates the fan power (L_{in}, L_{out}) to keep the chamber pressure ($\tilde{P}_{\mathrm{dw}}$) constant. The assignment functions can be any stochastic or deterministic function that can be expressed in the Turing-complete language that controls the chamber computer. This allows us to modify the causal structure underlying the chambers, by introducing additional effects of varying strength between variables. Although this yields a vast space of possible configurations, for the moment, we only provide datasets from the four configurations shown in Fig. 3.

In Supplementary Section III, we provide a detailed description of all the effects (that is, edges) in Fig. 3, based on the background knowledge and carefully designed experiments (Supplementary Figs. 2–9). Furthermore, Supplementary Section IV evaluates mechanistic models that describe some of the effects, ranging from simple natural laws to more complex models involving the technical specifications of the actual components. For the more complex processes in the chambers, such as image capture in the light tunnel or the effects on wind tunnel pressure, we provide approximate models with increasing degrees of fidelity. In Fig. 6c, we compare the output of some of these models to measurements gathered from the chambers.

Causal ground truth

For readers with a background in causal inference, the graphs in Fig. 3 may be reminiscent of causal graphical models[2,3,16]. In Supplementary Section V, we formalize a causal interpretation of the graphs and validate them with additional randomized experiments. In short, an edge $X \to Y$ signifies that an intervention on X will change the distribution of the subsequent measurements of Y. This interpretation allows us to treat the graphs shown in Fig. 3 as causal ground truths for a variety of causal inference tasks.

Under our interpretation, the absence of an edge between two variables does not preclude the existence of a causal effect between them. As with most real systems, effects between observed variables may exist beyond what we know or can validate through the procedures described in this paper, due to a lack of statistical power. Furthermore, there are confounding effects in which unmeasured variables simultaneously affect some of the variables in the chambers. For example, variations in the atmospheric pressure outside the chambers simultaneously affect all the barometric measurements. Supplementary Section V provides more details.

Case studies

We now show, through practical examples, how the chambers can be used to validate algorithms from a variety of fields. As a starting point, first, we provide a collection of datasets and set up tasks from a selection of research areas. Our choice is by no means exhaustive, and these case studies are intended as illustrations rather than comprehensive benchmarks. We describe each field and the corresponding tasks below and evaluate the performance of different algorithms, showing the results in Figs. 5 and 6.

For each case study, we provide a detailed description of the experimental procedure in the Methods, together with a well-documented code to reproduce the experiments in the paper repository at ref. 17. See the 'Data availability statement' for details of how to access the datasets.

Causal discovery

By offering a causal ground truth and the ability to carry out interventions, the chambers provide an opportunity to validate causal discovery algorithms[3,18–21], which aim to recover the cause-and-effect relationships from the data. The chambers provide data suited to validate a wide range of approaches, including those that rely on independent and identically distributed or time-series data[22] with and without instantaneous or lagged causal effects, and causal structures with and without cycles[23,24]. We consider the task of recovering the complete causal graph describing the effects in the system[1,25,26], and evaluate algorithms that take different types of data as the input: greedy equivalence search (GES)[27] for purely observational data, unknown-target interventional greedy sparsest permutation (UT-IGSP)[28] for interventional data with unknown targets and Peter-Clark momentary conditional independence (PCMCI+)[29] for time-series data. This constitutes an example selection of methods that is not exhaustive. Performance is measured by the recovery of the ground-truth graph (see the 'Causal ground truth' section). The results are shown in Fig. 5a. In line with their underlying assumptions, both GES and UT-IGSP recover the strong, linear effects from the light-source setting (R, G, B) to the light-sensor readings and drawn current. However, both methods struggle with the nonlinear effects of the polarizer angles (θ_1, θ_2) and the weak effects of the additional light-emitting diodes (LEDs; $L_{11}, \ldots, L_{32}$), which are apparent only in the cases when the light-source brightness is low or the polarizers are crossed (Supplementary Fig. 10). For the time-series data from the wind tunnel (task a3), PCMCI+ displays a low recall and performs similar to random guessing, despite the data matching the settings it is intended for.

Out-of-distribution generalization

By manipulating the chamber actuators and sensor parameters, we can induce distribution shifts in a controlled manner. This enables us not only to test the performance of the prediction and inference algorithms on datasets with a distribution that differs from the training distribution but also to investigate under which assumptions on the shifts such methods perform well[30–32]. As an illustration, we set up three simple tasks with different data modalities (Fig. 5b). The first consists of predicting the light-intensity reading $\tilde{I}_1$ from the other numeric variables of the light tunnel. We fit a simple linear regression with an increasing number of predictors and evaluate its predictive performance on data arising from interventions on the light-source intensity (R, G, B), sensor parameters $(T_2^I, T_1^V, T_2^V, T_3^V)$ and polarizer alignment (θ_1). For the second task, we predict the colour setting (R, G, B) of the light source from the images captured by the camera. We use a small convolutional neural network[33], which we evaluate on shifts induced by changing the distribution of colours, polarizer angles and camera parameters. The goal of the last task is to predict the hatch position H from the pressure curve $(\tilde{P}_{\mathrm{dw}})$ that results from applying a short impulse to the load L_{in} of the intake fan. We fit a simple feed-forward neural network and validate its performance on curves collected under different loads of the exhaust fan L_{out}, different barometer precision (O_{dw}) and from a barometer in a different position $(\tilde{P}_{\mathrm{up}})$. As expected, the performance of the methods degrades under distribution shifts. Even minute changes to the distribution of inputs, for example, due to an increase in the barometer oversampling rate $(O_{\mathrm{dw}} \leftarrow 8$; Extended Data Fig. 1), can make the multilayer perceptron in task b3 fail. Interestingly, the notion of causal invariance[34] predicts the drop in performance of some models. For example, the mean absolute error (MAE) incurred by predicting the training-set mean (that is, the empty model) remains constant across environments, except in those where the causal parents of the response (Fig. 3a) receive an intervention (that is, R, G, B in tasks b1 and b2). In task b1, the model that includes only causal parents $(\tilde{I}_1 \approx R, G, B)$ is the most stable across all the environments, whereas models that include additional (non-causal) variables achieve a better MAE in the training distribution but perform worse in environments in which these variables directly or indirectly receive an intervention.

Change point detection

Change point detection aims to identify abrupt changes or transitions in time-series data or its underlying data-generating process[35]. By manipulating actuators and sensor parameters, we can induce changes in the measurements of the affected sensors, providing real datasets with a known ground truth in terms of change points. To validate offline change point detection algorithms[35,36], we generate time-series data with smooth and abrupt changes of increasing difficulty. We evaluate the non-parametric change point detection algorithm changeforest[37], and the results are shown in Fig. 5c. As expected, the method correctly recovers all the change points in the deterministic time-series data of the actuator input L_{in}. For the affected sensors, the method successfully detects abrupt changes in the signal or its regime, but fails to detect more subtle changes, such as those with only a slight effect on the variance (for example, $\tilde{C}_{\mathrm{in}}$ or $\tilde{M}$ in Fig. 5c).

https://doi.org/10.1038/s42256-024-00964-x

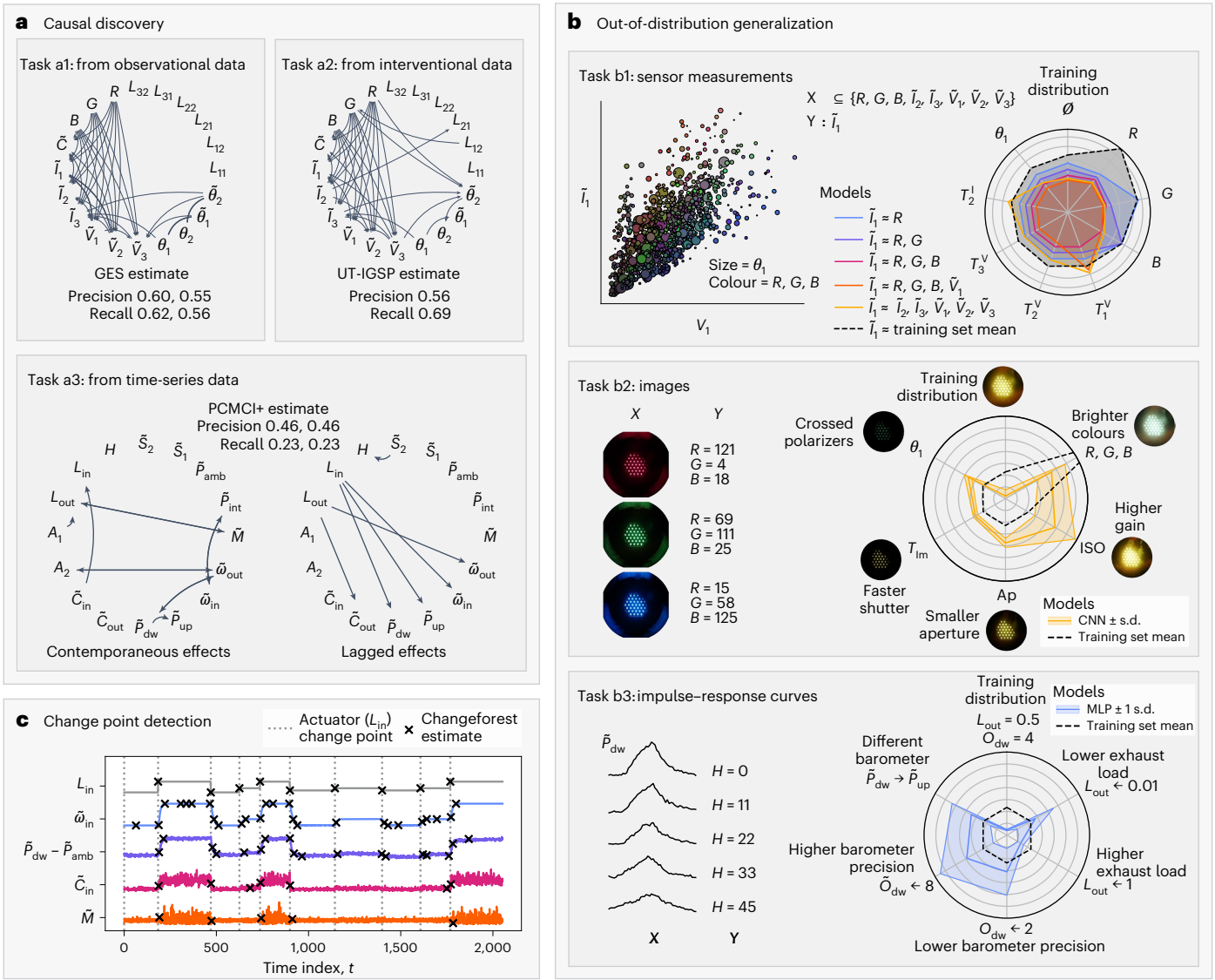


Fig. 5 | Validating algorithms using the chambers (part 1). a, Causal discovery. The tasks consist of recovering the causal graph using the observational data and interventional data from the light tunnel (tasks a1 and a2), as well as using the time-series data from the wind tunnel (task a3). We run a suitable method for each task (GES[27], UT-IGSP with hyperparameter tuning[28], and PCMCI+[29], respectively), and evaluate their performance in the recovery of the causal structure of the corresponding ground truth (see the 'Causal ground truth' section). GES and PCMCI+ return a set of 12 and 5 plausible graphs, respectively, encoded by a graph with undirected edges (see, for example, section 2.4 in ref. 27). For these methods, we show the precision and recall in the recovery of the directed ground-truth edges for the best-scoring (bold) and worst-scoring graph in each set. All the graphs returned by PCMCI+ attain the same scores, performing similarly to random guessing. **b**, Evaluating the out-of-distribution performance of regression methods. For each task, we try to predict a sensor measurement or actuator value (Y) from predictors (X) such as numeric measurements (task b1), images (task b2) or impulse–response curves (task b3). We evaluate the predictive performance of each method in terms of its MAE on a separate validation set from the training distribution and shifted distributions arising from manipulating the chamber variables. We display the MAE with spider charts, where each axis corresponds to a different setting. As a baseline, we show the MAE incurred when using the average of Y in the training set as prediction (black, dashed). For tasks b2 and b3, the MAE is averaged over 16 random initializations of the model, with error bands corresponding to ±1 standard deviation. **c**, Change point detection in sensor measurements. We change the intake fan load (L_{in}) at random time points and keep all other actuators and sensor parameters constant. Because the load affects all the displayed sensors, we take these time points as the ground truth (vertical dotted lines) and compare them with the output of the change point detection algorithm (black crosses). MLP, multilayer perceptron.

ICA

ICA is a family of techniques that treat data as a mixture of latent components and aim to discover a demixing transformation that can accurately recover them[38,39]. The linear variants of ICA[40,41] are well established, and recent developments in nonlinear ICA have cast it as a framework that holds potential for effectively tackling the challenge of disentanglement in complex data[6,38]. We propose tasks that consist of recovering (up to indeterminacies such as scaling) the values of independently set actuators from the measurements of the sensors they affect. As a starting point, we set up three tasks (Fig. 6a): recovering the light-source setting (R, G, B) from the light-intensity measurements ($\tilde{I}_1, \tilde{I}_2, \tilde{I}_3, \tilde{V}_1, \tilde{V}_2, \tilde{V}_3$), recovering the fan loads (L_{in}, L_{out}) and hatch position (H) from the barometric readings ($\tilde{P}_{dw}, \tilde{P}_{up}, \tilde{P}_{amb}, \tilde{P}_{int}$) and recovering the configuration of the light source and polarizers ($R, G, B, \theta_1, \theta_2$) from the image data of the light tunnel. The tasks display increasing difficulty in terms of the complexity and dimensionality of the mixing transformation. As the first baseline, we apply FastICA[41], which assumes a linear mixing function. Indeed, the method succeeds

https://doi.org/10.1038/s42256-024-00964-x

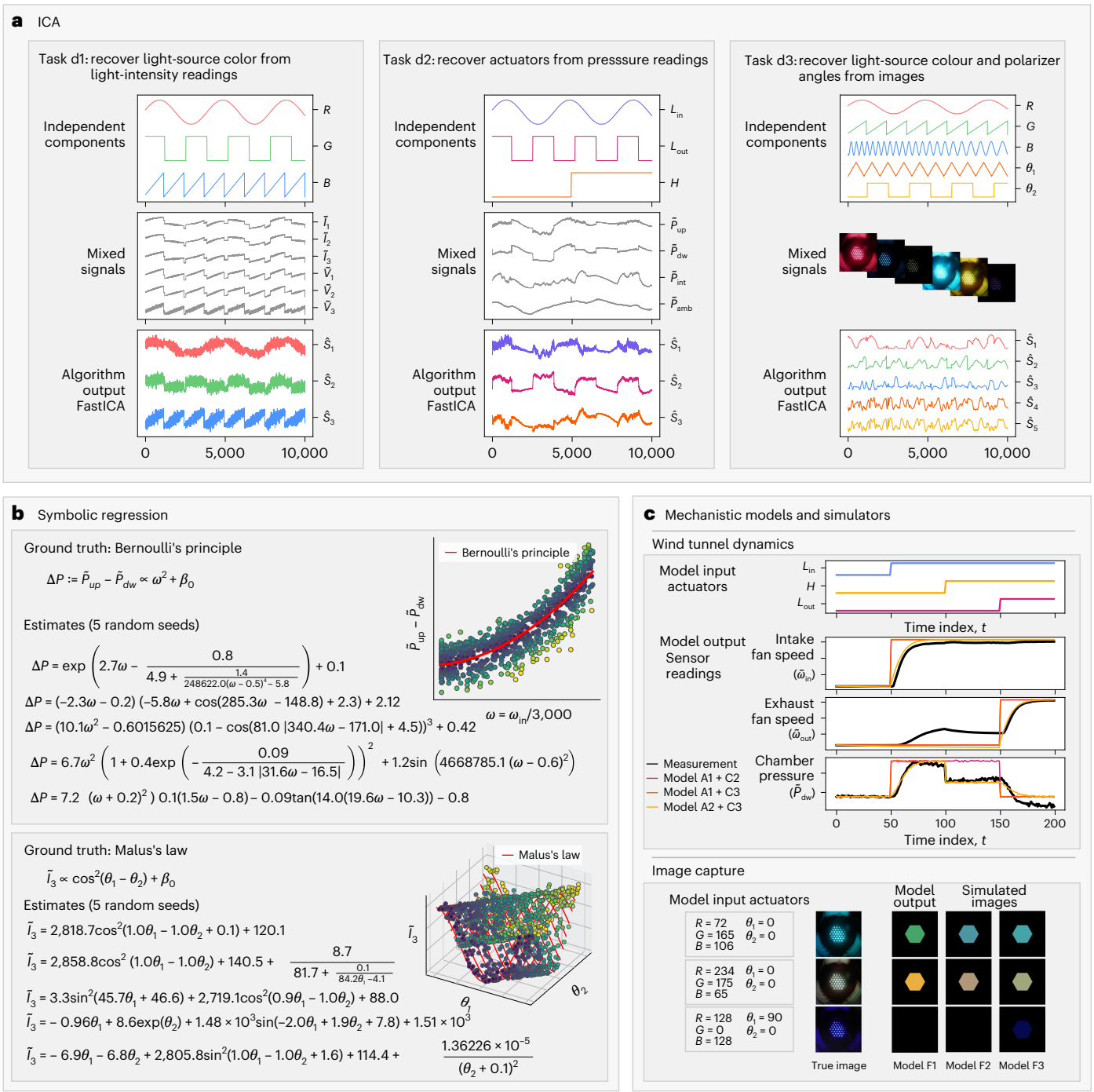


Fig. 6 | Validating algorithms using the chambers (part 2). a, Applying ICA to disentangle the actuator inputs from sensor readings (tasks d1 and d2) and image data (task d3). For each task, we show the actuator values (top), the resulting images and measurements (middle) and the sources recovered by the FastICA algorithm[41] (bottom). For each actuator, we show the recovered source with the highest Pearson correlation coefficient. **b**, Applying symbolic regression to recover (top) Bernoulli's principle from the difference in pressure at the upwind and downwind barometers, and (bottom) Malus's law from light-intensity measurements. We show the output of the method described elsewhere[43] for five runs with different random initializations. In the top panel, the colours correspond to the residual of the observation with respect to the red line and in the bottom panel, to the value of θ_2. **c**, Mechanistic models of the chambers to simulate sensor measurements from actuator inputs with varying degrees of fidelity. For a given set of inputs, we show the outputs of models describing the fan speeds and air pressure in the wind tunnel (top), and the image generation process of the light tunnel (bottom). We compare the model outputs with the images and measurements collected from the chambers (black lines). Higher model numbers imply more complex models with increased fidelity. The models are defined in Supplementary Section IV.

in estimating the actuator inputs for task d1 (Fig. 6), where the mixing function is approximately linear (Supplementary Section III.2). For the second task (d2), in which the effect of the actuators on the sensors is nonlinear (Supplementary Section IV.1.2), the method produces a distorted estimate of the actuators. For the third task (d3), in which the mixing function is both nonlinear and high dimensional, the method produces estimates in seemingly little agreement with the ground-truth signals.

Symbolic regression

Symbolic regression[12,42] aims to discover mathematical equations or expressions that best describe the underlying relationships in data, enabling interpretable and compact model representations. A common motivation is the automatic discovery of natural laws from the data[4]. Because simple natural laws effectively describe some of the relationships in the chambers, it is possible to provide symbolic regression tasks from real data, and evaluate the performance of such algorithms. As an example, we set up two tasks: recovering Bernoulli's principle, which relates the barometric measurements of the upwind and downwind barometers ($\tilde{P}_{\text{up}}, \tilde{P}_{\text{dw}}$); and Malus's law, which describes the effect of linear polarizers (θ_1, θ_2) on the light-intensity readings of the third sensor ($\tilde{I}_3, \tilde{V}_3$); more details can be found in Supplementary Sections IV.1.3 and IV.2.1, respectively. Bernoulli's principle provides a task with a simple ground-truth function but a low signal-to-noise ratio, whereas Malus's law provides a more complex function with weaker noise, representing two common challenges for symbolic regression algorithms. We apply the method described in ref. 43 and show the results of five runs in Fig. 6b. The estimated expressions depend strongly on the random initialization of the method, although all of them attain a similar R^2 score on the data (Extended Data Fig. 2). When we apply the method to synthetic data following Malus's law with added Gaussian noise, the dependence on random initialization disappears and the method returns the correct ground-truth expression in every run (Extended Data Fig. 2). This highlights a scenario in which synthetic benchmarks may be unreliable for estimating a method's performance in the real world.

Physics-informed machine learning

Physics-informed machine learning integrates physical laws or domain-specific knowledge into machine learning models to enhance their accuracy and generalizability[44]. To validate such approaches, in Supplementary Section IV, we provide mechanistic models of several processes in the chambers, derived from first principles. For each process, we consider models of increasing complexity, allowing us to simulate sensor measurements with varying degrees of fidelity. This provides a testbed for simulation-based inference[45] and approaches that exploit potentially mis-specified models for inference or generation[46–48]. As an illustration, in Fig. 6c, we compare measurements gathered from the chambers with the output of some of these models. In particular, we show the models describing the image capture process of the light tunnel, and the effects of fan loads ($L_{\text{in}}, L_{\text{out}}$) and hatch position ($H$) on other wind tunnel variables ($\tilde{P}_{\text{dw}}, \tilde{\omega}_{\text{in}}, \tilde{\omega}_{\text{out}}$). Their description, together with additional models and their outputs, are available in Supplementary Section IV. To facilitate building additional models and simulators, we provide the datasheets for every chamber component in Supplementary Section VI, detailing its technical specifications and physical properties.

Discussion

We have constructed two devices to collect real-world datasets from well-understood but non-trivial physical systems. The devices provide a testbed beyond the simulated data for a variety of empirical inference algorithms in the broad field of AI. To illustrate their use, we have gathered an initial collection of datasets and used them to perform small case studies in different fields.

The case studies are intended to showcase the flexibility of the chambers in setting up validation tasks; providing exhaustive benchmarks is beyond the scope of this work. However, the mixed performance of algorithms in the case studies suggests that, although limited, they can already serve as useful benchmarks for these fields. In some cases, the shortcomings of the methods can be attributed to their underlying assumptions (for example, tasks a1, a2, d2 and d3). In others, such as tasks e and a3, a mismatch is highlighted between performance on synthetic and real data, which can lead to an overconfident assessment of a method's capabilities. Task b shows that the chambers provide a principled environment to study phenomena such as causal invariance or the sensitivity of neural networks to small shifts in the distribution of their inputs.

We believe the presented chambers can be used for applications that go beyond the ones we have considered. In particular, the digital control of the chambers makes it possible to validate a variety of active learning, reinforcement learning and control algorithms.

The chambers are complementary to well-motivated, complex simulators of real phenomena. On one hand, such simulators allow us to approximate complex systems that are the intended application targets, such as mechanisms of the global climate or gene regulatory networks with hundreds of variables and interactions. On the other hand, it can be difficult (or impossible) to judge if the assumptions used to build these simulators will hold in the real world, and—more importantly—how their violation will affect an algorithm when we use it on real rather than simulated data. Well-understood systems like the chambers provide real-world data without relying on computer simulations and their models, as well as simplifying assumptions. However, the requirement of providing a reliable ground truth necessarily limits the chambers' complexity and size. Therefore, the success of an algorithm on the chambers may not necessarily transfer to larger and more complex systems.

Our aim is that the chambers become a sanity check for algorithms designed to work in a variety of situations. Failures in these testbeds can indicate potential shortcomings in applications to more complex systems. This will allow researchers to test and refine algorithms and methods, and consider fundamental assumptions.

We make all the datasets collected from the chambers publicly available, including those used in the 'Case studies' section. Researchers can access them at https://causalchamber.org and through the Python causalchamber package available at https://pypi.org/project/causalchamber/ (see the 'Data availability statement' for more details). We will continue to expand this dataset repository, and we are open to suggestions of additional experiments that may prove interesting—please reach out to the corresponding author.

In ref. 15, we provide the blueprints and code to allow other researchers to build their own chambers. We hope these resources can be used as a starting point to build chambers around other well-understood systems that prove valuable for the validation of AI methodology.

Methods

Here we provide a brief description of the experimental setup for each case study described in the 'Case studies' section, together with a link to the corresponding datasets at https://causalchamber.org/, and to the corresponding code in the paper repository in ref. 17.

Case study: causal discovery

All the methods we evaluate in this case study return a directed acyclic graph (or a set of them) as an estimate. Given a single directed acyclic graph estimate $\hat{G} := (V, \hat{E})$ and a ground-truth graph $G^\star := (V, E^\star)$, we compute the precision P and recall R in terms of directed edge recovery as

$$P := \frac{\hat{E} \cap E^\star}{|\hat{E}|} \text{ and } R := \frac{\hat{E} \cap E^\star}{|E^\star|}, \tag{1}$$

where $\hat{E}$ and $E^\star$ are the sets of directed edges in $\hat{G}$ and $G^\star$, respectively. If a method outputs several directed acyclic graphs, we compute P and R for each element in this set.

Task a1: observational data. As input for GES, we take 10,000 observations from a subset of the variables (Fig. 5) in the uniform_reference experiment of the lt_interventions_standard_v1 dataset (https://github.com/juangamella/causal-chamber/tree/master/datasets/lt_interventions_standard_v1) and used the accompanying code

(https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/causal_discovery_iid.ipynb). As score for the algorithm, we use the bayesian information criterion (BIC) score with a Gaussian likelihood. GES returns the Markov equivalence class of the estimated data-generating graph, and for each graph, we compute the corresponding precision and recall in the recovery of the edges in the ground-truth graph.

Task a2: interventional data. We consider the same subset of variables as for task a1, taking data from several experiments in the lt_interventions_standard_v1 dataset (https://github.com/juangamella/causal-chamber/tree/master/datasets/lt_interventions_standard_v1) as input for UT-IGSP[28]. As 'observational data', we take the 10,000 observations from the uniform_reference experiment. As 'interventional data', we take 1,000 observations from each experiment in which the considered variables receive an intervention; see the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/causal_discovery_iid.ipynb) for the experiment names. For the conditional independence and invariance tests, we use the default Gaussian tests implemented in the Python package of UT-IGSP, and run the algorithm at different significance levels $(\alpha, \beta) \in [10^{-4}, 10^{-2}]^2$. We show the result for $\alpha = 0.008$ and $\beta = 0.009$, which performs best in terms of both precision and recall (equation (1)).

Task a3: time-series data. As input to PCMCI+[29], we take 10,000 observations from a subset of the variables in the actuators_random_walk_1 experiment of the wt_walks_v1 dataset (https://github.com/juangamella/causal-chamber/tree/master/datasets/wt_walks_v1) and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/causal_discovery_time.ipynb). We run the method with partial correlation tests at significance level $\alpha = 1 \times 10^{-2}$ and a maximum of ten lags. From the resulting estimate, we drop edges from a variable to itself and edges for which orientation conflicts arise. We compute the precision and recall (equation (1)) for each of the two graphs in the resulting equivalence class.

Case study: out-of-distribution generalization

Task b1: regression from sensor measurements. We use the data from several experiments in the lt_interventions_standard_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_interventions_standard_v1) and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ood_sensors.ipynb). We begin by splitting the observations from the uniform_reference experiment into a training set (100 observations) and a validation set (1,000 observations, shown with $\emptyset$ in the spider plot for task b1 (Fig. 5b)). As additional validation sets (1,000 observations each), we select experiments in which the variables $R, G, B, T_1^V, T_2^V, T_3^V, T_2^I, \theta_1$ receive an intervention; the accompanying code provides the experiment names. These validation sets correspond to the additional axes in the spider plot of task b1 in Fig. 5b. On the training set, we fit linear models with the intercept using ordinary least squares, with response $\tilde{I}_1$ and different sets of predictors: $\{R\}$, $\{R, G\}$, $\{R, G, B\}$, $R, G, B, \tilde{V}_1$ and $\{\tilde{I}_2, \tilde{I}_3, \tilde{V}_1, \tilde{V}_2, \tilde{V}_3\}$. As a baseline, we consider the model that predicts the average of $\tilde{I}_1$ in the training set. For each resulting model, we compute the MAE on each of the validation sets. The additional scatter plot for task b1 in Fig. 5b corresponds to the pooled data across all the validation sets.

Task b2: regression from images. We use the images from the lt_color_regression_v1 datasets (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_color_regression_v1) and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ood_images.ipynb), at a size of 100 × 100 pixels. We split the data from the reference experiment into a training and validation set (9,000 and 500 observations, respectively). As additional validation sets, we take those arising from shifts in the distribution of the response R, G, B (bright_colors experiment) and from interventions on the parameters of the camera; the accompanying code provides the experiment names. We subsample each of the additional validation sets to a size of 500 observations. As a regression model, we use a small LeNet-like convolutional neural network[49]; the code provides more details. As a loss function, we use the mean-squared error in predicting the light-source settings R, G, B, which we minimize using the stochastic gradient descent. We fit the model a total of 16 times, each with a different random initialization of the network weights. For each resulting model, we compute the MAE on each validation set, and plot the results in task b2 in Fig. 5b. As a baseline, we consider the model that predicts the average of R, G, B in the training set.

Task b3: regression from impulse–response curves. We use the data from several experiments in the wt_intake_impulse_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/wt_intake_impulse_v1), corresponding to different settings of the exhaust load L_{out} and oversampling rates O_{dw} of the downwind barometer; the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ood_impulses.ipynb) provides the experiment names. We split the data from the load_out_0.5_osr_downwind_4 experiment into a training and validation set (4,000 and 900 observations, respectively). As a regression model, we use a multilayer perceptron with an input layer of size 50 (the impulse length), an output layer of size 1 and two additional hidden layers with 200 neurons and rectified linear unit activations. As a loss function, we use the mean-squared error in predicting the hatch position H, and train the model using stochastic gradient descent. We fit the model a total of 16 times, each with a different random initialization of the network weights. For each resulting model, we compute the MAE on validation sets from the training distribution and additional experiments. Each corresponds to the different axes in the spider plot for task b3 in Fig. 5b. As a baseline, we consider the model that predicts the average of H in the training set.

Case study: change point detection

We take the data from the load_in_seed_9 experiment in the wt_changepoints_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/wt_changepoints_v1), and apply the changeforest algorithm[37] to each of the time-series data, namely, L_{in}, $\bar{\omega}_{\text{in}}$, $\bar{P}_{\text{dw}} - \bar{P}_{\text{amb}}$, $\bar{C}_{\text{in}}$, $\bar{M}$. For the algorithm, we use the 'random_forest' method and default hyperparameters; the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/changepoints.ipynb) provides the details. As the ground truth for the change points (Fig. 5c, vertical grey lines), we take the time points in which L_{in} is set to a new level. In all the datasets collected from the chambers, the column intervention takes a value of 1 for the first measurement after an intervention on any of the chamber variables.

Case study: ICA

Task d1: recovering light-source colour. We use the color_mix experiment from the lt_walks_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_walks_v1) and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ica.ipynb). As an input to the FastICA algorithm[41], we take the light-intensity measurements $\tilde{I}_1, \tilde{I}_2, \tilde{I}_3, \tilde{V}_1, \tilde{V}_2, \tilde{V}_3$, to which we first apply a whitening transformation. We run the algorithm with six components (sources). For each ground-truth source (R, G, B), we show the recovered signal with the highest Pearson correlation coefficient (in absolute value).

Task d2: recovering fan loads and hatch position. We use the loads_hatch_mix_slow experiment from the wt_walks_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/wt_walks_v1) and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ica.ipynb). As an input to the FastICA algorithm[41], we take the barometric pressure measurements $\tilde{P}_{dw}, \tilde{P}_{up}, \tilde{P}_{amb}, \tilde{P}_{int}$, to which we first apply a whitening transformation. We run the algorithm with four components (sources). For each ground-truth source (L_{in}, L_{out}, H), we show the recovered signal with the highest Pearson correlation coefficient (in absolute value).

Task d3: recovering actuators from images. As an input to the FastICA algorithm[41], we use the images from the actuator_mix experiment in the lt_camera_walks_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_camera_walks_v1), at a size of 50 × 50 pixels, and used the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/ica.ipynb). We first flatten the images, such that each pixel becomes an input variable, applying a whitening transformation as an additional preprocessing step. We run the algorithm with five components (sources). For each ground-truth source (R, G, B, θ_1, θ_2), we show the recovered signal with the highest Pearson correlation coefficient (in absolute value).

Case study: symbolic regression

We use a pretrained version of the model described in ref. 43; the code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/symbolic_regression.ipynb) provides more details. For both tasks, we use the same hyperparameters as that in the demonstration provided at https://github.com/facebookresearch/symbolicregression/blob/main/Example.ipynb and run the algorithm with five different random initializations. As input for the first task, we randomly sample 1,000 observations from the random_loads_intake experiment in the wt_bernoulli_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/wt_bernoulli_v1); for the second task, we use 1,000 observations from the white_255 experiment in the lt_malus_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_malus_v1). We show the estimated expressions in Fig. 6b, rounding the constants to one decimal place.

Case study: mechanistic models

We compare the output of some of the models defined in Supplementary Section IV with actual measurements collected from the chamber. For the wind tunnel models, we use the steps experiment from the wt_test_v1 (https://github.com/juangamella/causal-chamber/tree/main/datasets/wt_test_v1), setting their parameters to the values suggested in Supplementary Section IV.1. For the models of the image capture process of the light tunnel, we take the images from the palette experiment in the lt_camera_test_v1 dataset (https://github.com/juangamella/causal-chamber/tree/main/datasets/lt_camera_test_v1) and use the parameters suggested in Supplementary Section IV.2.2. All the models are implemented in the Python package available at https://github.com/juangamella/causal-chamber#mechanistic-models; the accompanying code (https://github.com/juangamella/causal-chamber-paper/blob/main/case_studies/mechanistic_models.ipynb) provides examples.

Data availability

All datasets are available via Github at https://github.com/juangamella/causal-chamber (ref. 15) along with instructions for accessing them. The identifier for each dataset used in the case studies is specified in the Methods. A Python API for directly downloading and importing the datasets into your code is provided in the Python package causalchamber, available at https://pypi.org/project/causalchamber/.

Code availability

The code to reproduce the case studies and figures is available at https://github.com/juangamella/causal-chamber-paper (refs. 17,50).

References

1. Spirtes, P., Glymour, C., Scheines, R. & Heckerman, D. *Causation, Prediction, and Search* (MIT Press, 2000).
2. Pearl, J. *Causality* (Cambridge Univ. Press, 2009).
3. Peters, J., Janzing, D. & Schölkopf, B. *Elements of Causal Inference: Foundations and Learning Algorithms* (MIT Press, 2017).
4. Schmidt, M. & Lipson, H. Distilling free-form natural laws from experimental data. *Science* **324**, 81–85 (2009).
5. La Cava, W. et al. Contemporary symbolic regression methods and their relative performance. In *35th Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)* (2021).
6. Locatello, F. et al. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proc. 36th International Conference on Machine Learning* 4114–4124 (PMLR, 2019).
7. Schölkopf, B. et al. Toward causal representation learning. *Proc. IEEE* **109**, 612–634 (2021).
8. Koh, P. W. et al. WILDS: a benchmark of in-the-wild distribution shifts. In *Proc. 38th International Conference on Machine Learning*, 5637–5664 (PMLR, 2021).
9. Gamella, J. L. & Heinze-Deml, C. Active invariant causal prediction: experiment selection through stability. *Adv. Neural Inf. Process. Syst.* **33**, 15464–15475 (2020).
10. Göbler, K. et al. causalAssembly: generating realistic production data for benchmarking causal discovery. In *Causal Learning and Reasoning* 609–642 (PMLR, 2024).
11. Cheng, Y. et al. CausalTime: realistically generated time-series for benchmarking of causal discovery. In *The Twelfth International Conference on Learning Representations* (2023).
12. Udrescu, Silviu-Marian & Tegmark, M. AI Feynman: a physics-inspired method for symbolic regression. *Sci. Adv.* **6**, eaay2631 (2020).
13. Greenfield, A., Madar, A., Ostrer, H. & Bonneau, R. Dream4: combining genetic and dynamic information to identify biological networks and dynamical models. *PLoS ONE* **5**, e13397 (2010).
14. Tu, R., Zhang, K., Bertilson, B., Kjellstrom, H. & Zhang, C. Neuropathic pain diagnosis simulator for causal discovery algorithm evaluation. *Adv. Neural Inf. Process. Syst.* **32**, 6955 (2019).
15. Gamella, J. L. Dataset repository for the causal chambers. *GitHub* https://github.com/juangamella/causal-chamber (2024).
16. Lauritzen, S. L. Causal inference from graphical models. *Monogr. Stat. Appl. Prob.* **87**, 63–108 (2001).
17. Gamella J. L. Experiments repository for the causal chambers. *GitHub* https://github.com/juangamella/causal-chamber-paper (2024).
18. Pearl, J. Causal inference in statistics: an overview. *Statist. Surv.* **3**, 96–146 (2009).
19. Glymour, C., Zhang, K. & Spirtes, P. Review of causal discovery methods based on graphical models. *Front. Genet.* **10**, 524 (2019).
20. Heinze-Deml, C., Maathuis, M. H. & Meinshausen, N. Causal structure learning. *Annu. Rev. Stat. Appl.* **5**, 371–391 (2018).
21. Mooij, J. M., Peters, J., Janzing, D., Zscheischler, J. & Schölkopf, B. Distinguishing cause from effect using observational data: methods and benchmarks. *J. Mach. Learn. Res.* **17**, 1–102 (2016).
22. Runge, J. Causal network reconstruction from time series: from theoretical assumptions to practical estimation. *Chaos* **28**, 075310 (2018).
23. Bongers, S., Forré, P., Peters, J. & Mooij, J. M. Foundations of structural causal models with cycles and latent variables. *Ann. Stat.***49**, 2885–2915 (2021).

24. Claassen, T. & Mooij, J. M. Establishing Markov equivalence in cyclic directed graphs. In *Proc. 39th Conference on Uncertainty in Artificial Intelligence* 433–442 (PMLR, 2023).
25. Shimizu, S., Hoyer, P. O., Hyvärinen, A. & Kerminen, A. A linear non-Gaussian acyclic model for causal discovery. *J. Mach. Learn. Res.* **7**, 2003–2030 (2006).
26. Spirtes, P., Meek, C. & Richardson, T. An algorithm for causal inference in the presence of latent variables and selection bias. *Comput. Causation Discov.* **21**, 211–252 (1999).
27. Chickering, D. M. Optimal structure identification with greedy search. *J. Mach. Learn. Res.* **3**, 507–554 (2002).
28. Squires, C., Wang, Y. & Uhler, C. Permutation-based causal structure learning with unknown intervention targets. In *Proc. 36th Conference on Uncertainty in Artificial Intelligence* 1039–1048 (PMLR, 2020).
29. Runge, J. Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets. In *Proc. 36th Conference on Uncertainty in Artificial Intelligence* 1388–1397 (PMLR, 2020).
30. Nagarajan, V., Andreassen, A. & Neyshabur, B. Understanding the failure modes of out-of-distribution generalization. In *Proc. 8th International Conference on Learning Representations* (2020).
31. Geirhos, R. et al. Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2**, 665–673 (2020).
32. Rothenhäusler, D., Meinshausen, N., Bühlmann, P. & Peters, J. Anchor regression: heterogeneous data meet causality. *J. R. Stat. Soc. B* **83**, 215–246 (2021).
33. Fukushima, K. Neocognitron: a hierarchical neural network capable of visual pattern recognition. *Neural Netw.* **1**, 119–130 (1988).
34. Peters, J., Bühlmann, P. & Meinshausen, N. Causal inference by using invariant prediction: identification and confidence intervals. *J. R. Stat. Soc. B* **78**, 947–1012 (2016).
35. Truong, C., Oudre, L. & Vayatis, N. Selective review of offline change point detection methods. *Signal Process.* **167**, 107299 (2020).
36. Aminikhanghahi, S. & Cook, D. J. A survey of methods for time series change point detection. *Knowl. Inf. Syst.* **51**, 339–367 (2017).
37. Londschien, M., Bühlmann, P. & Kovács, S. Random forests for change point detection. *J. Mach. Learn. Res.* **24**, 1–45 (2023).
38. Hyvärinen, A., Khemakhem, I. & Morioka, H. Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning. *Patterns* **4**, 100844 (2023).
39. Hyvärinen, A., Karhunen, J. & Oja, E. *Independent Component Analysis* (Wiley Interscience, 2001).
40. Hyvärinen, A. & Oja, E. Independent component analysis: algorithms and applications. *Neural Netw.* **13**, 411–430 (2000).
41. Hyvärinen, A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. Neural Netw.* **10**, 626–634 (1999).
42. Cranmer, M. et al. Discovering symbolic models from deep learning with inductive biases. *Adv. Neural Inf. Process. Syst.* **33**, 17429–17442 (2020).
43. Kamienny, Pierre-Alexandre, d'Ascoli, Stéphane, Lample, G. & Charton, François End-to-end symbolic regression with transformers. *Adv. Neural Inf. Process. Syst.* **35**, 10269–10281 (2022).
44. Karniadakis, G. E. et al. Physics-informed machine learning. *Nat. Rev. Phys.* **3**, 422–440 (2021).
45. Cranmer, K., Brehmer, J. & Louppe, G. The frontier of simulation-based inference. *Proc. Natl Acad. Sci. USA* **117**, 30055–30062 (2020).
46. Takeishi, N. & Kalousis, A. Physics-integrated variational autoencoders for robust and interpretable generative modeling. *Adv. Neural Inf. Process. Syst.* **34**, 14809–14821 (2021).
47. Wehenkel, A. et al. Robust hybrid learning with expert augmentation. In *Transactions on Machine Learning Research* (2023).
48. Yin, Y. et al. Augmenting physical models with deep networks for complex dynamics forecasting. *J. Stat. Mech. Theory Exp.***2021**, 124012 (2021).
49. LeCun, Y., Bottou, L. éon, Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2324 (1998).
50. Gamella, J. L. juangamella/causal-chamber-paper: v1.0.0-alpha. *Zenodo* https://doi.org/10.5281/zenodo.14050466 (2024).

Acknowledgements

We thank all reviewers for their constructive and insightful comments. We thank M. Cherep, C. Fuchs, K. Göbler, C. Heinze-Deml, J. Jakobsen, N. Pfister, A. Wehenkel and T. Windisch for valuable discussions and comments on the manuscript. We also thank N. Stolz for his help with the design of the polarizer frames of the light tunnel, and C. Linares and H. Börjesson for their help with the diagrams and photographs of the chambers. J.L.G. and P.B. have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 786461).

Author contributions

J.L.G. conceived the study, designed and built the chambers, collected the datasets from the chambers, implemented the case studies, and prepared the manuscript and supplementary information. J.P. and P.B. supervised the project, discussed the analysis of the case studies and edited the manuscript.

Funding

Open access funding provided by Swiss Federal Institute of Technology Zurich.

Competing interests

The authors declare no competing interests.

Additional information

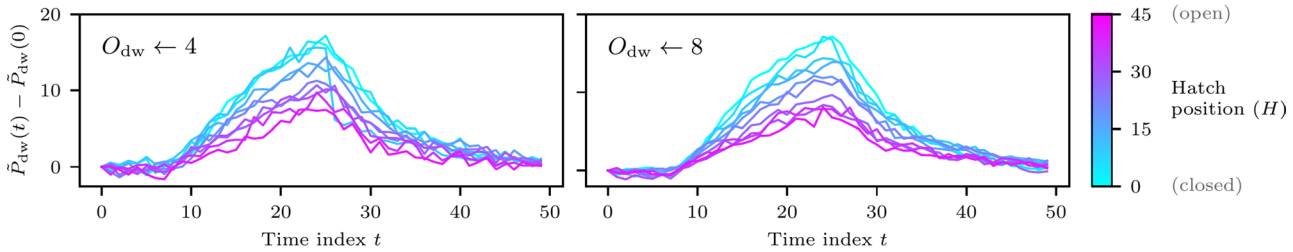
Extended data is available for this paper at https://doi.org/10.1038/s42256-024-00964-x.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-024-00964-x.

Correspondence and requests for materials should be addressed to Juan L. Gamella.

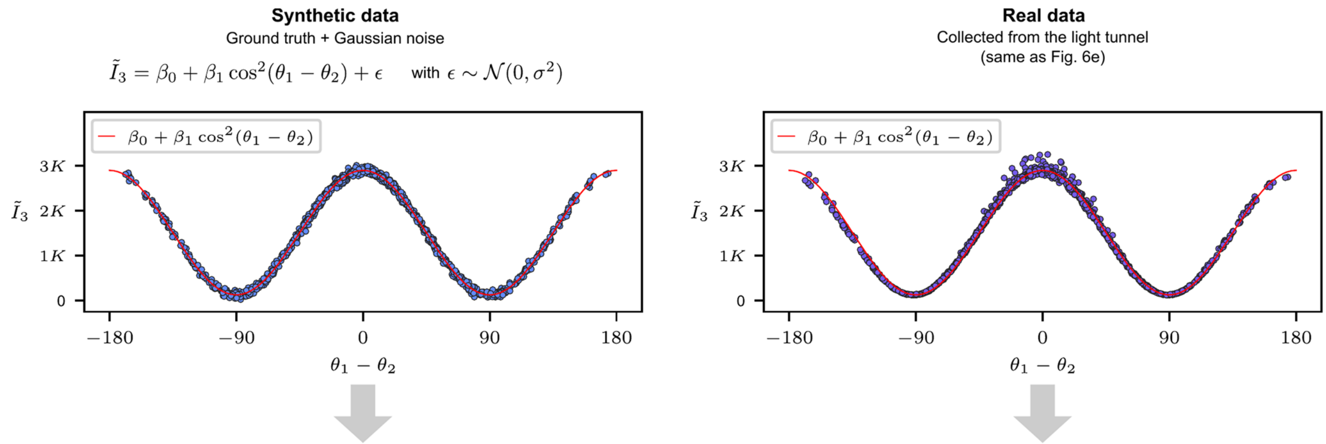
Peer review information *Nature Machine Intelligence* thanks Jakob Zeitler, Claudia Shi and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

https://doi.org/10.1038/s42256-024-00964-x



Extended Data Fig. 1 | Distribution shift in the pressure-curve dataset. Visualization of the pressure-curve dataset used in task b3 of Fig. 5. We show 10 curves picked at random for different positions of the hatch, under the oversampling rate seen during training (Odw ← 4, left) and the one used as an out-of-distribution test (Odw ← 8, right). The oversampling rate determines how many barometer readings are averaged to produce a single measurement (see Supplementary Table 1), increasing or decreasing its precision. While the effect on the signal is almost indistinguishable, the change is enough to cause the MLP in task b3 to fail in its out-of-distribution predictions.

 https://doi.org/10.1038/s42256-024-00964-x



Estimates (5 random seeds) — synthetic data

$$(R^2 = 0.99) \quad \tilde{I}_3 = 2887.1 \cos^2(1.0\theta_1 - 1.0\theta_2) + 80.2$$
$$(R^2 = 0.99) \quad \tilde{I}_3 = 2887.1 \cos^2(1.0\theta_1 - 1.0\theta_2) + 80.2$$
$$(R^2 = 0.99) \quad \tilde{I}_3 = 2887.1 \cos^2(1.0\theta_1 - 1.0\theta_2) + 80.2$$
$$(R^2 = 0.99) \quad \tilde{I}_3 = 2887.1 \cos^2(1.0\theta_1 - 1.0\theta_2) + 80.2$$
$$(R^2 = 0.99) \quad \tilde{I}_3 = 2887.1 \cos^2(1.0\theta_1 - 1.0\theta_2) + 80.2$$

Estimates (5 random seeds) — real data

$$(R^2 = 0.99) \quad \tilde{I}_3 = 2818.7 \cos^2(1.0\theta_1 - 1.0\theta_2 + 0.1) + 120.1$$
$$(R^2 = 0.98) \quad \tilde{I}_3 = 2858.8 \cos^2(1.0\theta_1 - 1.0\theta_2) + 140.5 + \frac{8.7}{81.7 + \frac{0.1}{84.2\theta_1 - 4.1}}$$
$$(R^2 = 0.99) \quad \tilde{I}_3 = -6.9\theta_1 - 6.8\theta_2 + 2805.8 \sin^2(1.0\theta_1 - 1.0\theta_2 + 1.6) + 114.4 + \frac{1.36226 \cdot 10^{-5}}{(\theta_2 + 0.1)^2}$$
$$(R^2 = 0.97) \quad \tilde{I}_3 = 3.3 \sin^2(45.7\theta_1 + 46.6) + 2719.1 \cos^2(0.9\theta_1 - 1.0\theta_2) + 88.0$$
$$(R^2 = 0.98) \quad \tilde{I}_3 = -0.96\theta_1 + 8.6 \exp(\theta_2) + 1.48 \cdot 10^3 \sin(-2.0\theta_1 + 1.9\theta_2 + 7.8) + 1.51 \cdot 10^3$$

Extended Data Fig. 2 | Symbolic regression on synthetic vs. real data. Estimated expressions and their R2 scores when we apply the symbolic regression method from Fig. 6b to the real data from the light tunnel (right), and synthetic data (left) following Malus' law (see Supplementary Material IV.2.1). The synthetic data is produced by fitting the law to the data and adding Gaussian noise to simulate sensor noise. For the real data, the estimated expression varies with the random initialization of the method, whereas for the synthetic data, the method recovers the ground-truth expression in each run. This phenomenon does not carry over to the task of recovering Bernoulli's principle, where the method output is highly variable for both the synthetic and real data (see Supplementary Figure 1).