Learning Stable Causal Explanations from Deep Neural Networks

Sara Pernille Jensen
(Dated: March 14, 2024)
(Github repository: https://github.com/SaraPJensen/FYS9426)

CONTENTS

1. INTRODUCTION

2. THEORY

2.1. Machine learning

2.1.1. Statistical Learning Theory

Statistical learning theory is the underlying theory for most machine learning methods. In essence, it is the theory of how one might learn to make optimal statistical predictions based on data. Assume we have a dataset consisting of input variables $\mathbf{X} \in \mathbb{R}^F$ and output variables $\mathbf{Y} \in \mathbb{R}^M$. We assume that there is some true conditional probability distribution $P(\mathbf{Y}|\mathbf{X})$ underlying the data. If this was known, given any input $\mathbf{X}$, this conditional distribution would provide us with the best possible prediction. This is also known as the *optimal Bayes decision rule*, and it can be shown that no other decision rule will give a lower error on the predictions [9]. In the case of a deterministic system where $\mathbf{Y}$ is only dependent on $\mathbf{X}$, this would always give us the right prediction.

The problem is that in cases where the number of input- and output-features F and M, or the number of datapoints N, are large, calculating this conditional probability distribution becomes extremely computationally heavy, to the point where it is impossible in practice. This is known as the *curse of dimensionality*. What one can do instead, then, is to find a function f_θ with parameters θ, whose predictions given $\mathbf{X}$ approximates the true distribution $P(\mathbf{Y}|\mathbf{X})$. The model's conditional probability distribution is $P_\theta(\mathbf{Y}|\mathbf{X})$, and the goal is that $P_\theta(\mathbf{Y}|\mathbf{X}) \rightarrow P(\mathbf{Y}|\mathbf{X})$. In practice this means that given any $\mathbf{X}$ the function produces the same output $\mathbf{Y}$ as the original dataset. Note that many different combinations of parameters θ may give rise to the same probability distribution.

There are many different methods for approximating such a function f_θ, including a range of different methods from machine learning. For our current purposes, only deep neural networks will be considered, as this is likely the most popular and versatile method used.

2.1.2. Deep Neural Networks

We still assume the dataset of independent variables $\mathbf{X} \in \mathbb{R}^F$ with corresponding dependent variables $\mathbf{Y} \in \mathbb{R}^M$. To make the optimal prediction of $\mathbf{Y}$ given $\mathbf{X}$, we train a network $\mathcal{F}$ with parameters θ. For the current purposes, only a simple feed forward neural network will be considered as the network $\mathcal{F}$. The trainable parameters θ are then the weights and the biases of the network. As for the dataset, we assume that all the data is drawn from the same underlying probability distribution, and we split this into a training set and a test set, where only the training data is presented to the network during training. Given the finiteness of the dataset, the model is thus necessarily trained on data from a certain *domain*, as well as from a certain *distribution*.

As the mathematical theory behind such networks have been elaborately outlined in a number of reference works, I simply refer the reader to e.g., [13] for

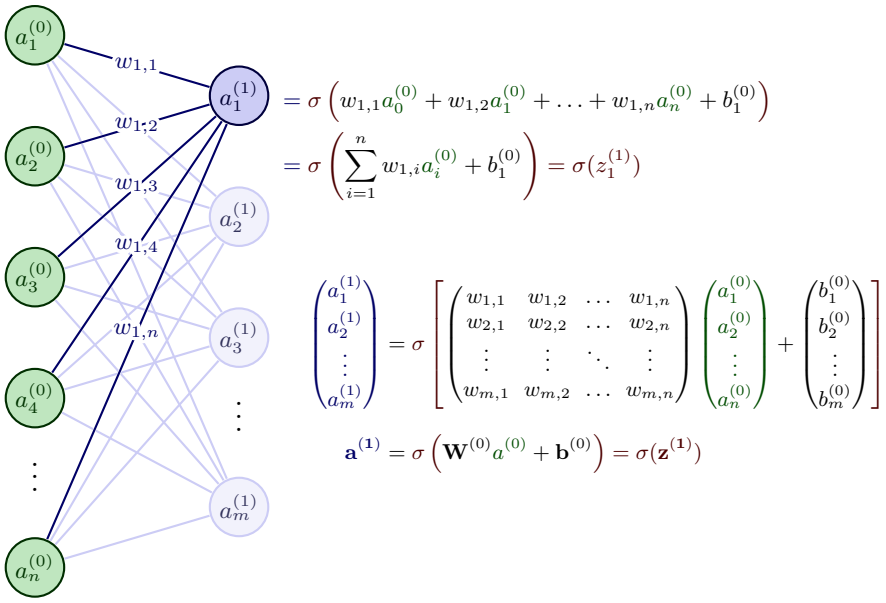

Figure 1: A feed forward neural network showing the input layer and the first hidden layer along with the corresponding calculations. Figure adapted from [12]. Here σ is the Sigmoid function.

an outline and derivation of the equations used in the feed-forward and back-propagation stages. The

Since the data considered here take continuous values, the cost function used is simple mean squared error (MSE).

2.1.3. Model Performance

We can identify two desirable traits of the trained machine learning model: *reliability* and *robustness*. Reliability is what is typically tested, and is done by testing the performance of the model on the test-data coming from the same domain and probability distribution as the model was trained on. This should be reasonably close to the final performance of the model on the training data. If the test error is significantly higher, this means that the model does *not* generalise to unseen data, so has not learned the correct distribution $P(\mathbf{Y}|\mathbf{X})$. Consequently, one cannot expect the model to give correct predictions on new data, so the model is unreliable. Good performance on test data is thus the bare minimum we must expect from a trained machine learning model for it to be the least bit useful. This is also what is usually tested and referred to in the research.

Another potentially desirable performance feature is the model's robustness. This describes how well the model works on different, but related, tasks. This difference may involve a slight change in domain or a change in the underlying probability distribution of the data. A robust model thus has a wider

feed forward stage is illustrated in 1, with the corresponding equation.

range of applicability, and must thus have learned some structure underlying both the initial and the new domain or distribution.

How different one can expect the domain to be without a reduction in the performance is of course context-dependent, and is a question which applies to the use of other models in science as well. For example, one cannot expect the model of simple harmonic motion of a ball on a string to hold in a storm. In general, ML models are less robust than what one might expect and desire [8, 15, 20].

Thus, the two metrics for performance considered here are:

Reliability: Performance on unseen data taken from the same domain and distribution.

Robustness: Performance on unseen data taken from either (or both) a new domain or from a different distribution.

2.1.4. Black Box Modelling

Assuming we have a reasonably reliable model, what can we learn from this other than the predictions on new data themselves? Not necessarily that much, since the only information easily available to

the user are the inputs and their corresponding outputs. For this reason, such models are often called *black box models*, and are described as *opaque* [3, 17]. It is not clear for a user *why* or *how* a given output follows from a given input.

One might respond, then, that it is in theory perfectly possible to spell out the entire final equation which the model calculates. However, this does not seem to solve the problem, as discussed in [2]. Whereas traditional scientific models have mathematical terms that come with a clear physical interpretation, the equations produced by such a spelling out of the ML model would not. Or at least, it would not do so in terms of the same physical variables that we use. For example, assume we train a model to learn the mapping from amplitude A, the spring constant k, mass m and time t to height y, frequency f and period T, given by

$$\begin{aligned} y =& A\sin(\sqrt{\frac{k}{m}}t) \\ \omega =& \sqrt{\frac{k}{m}} \\ f =& \frac{\omega}{2\pi} \\ T =& \frac{1}{f} \end{aligned}$$

Assuming we have a deep network with a reasonable number of nodes in each layer, the equation calculated could consist of arbitrarily many terms, and to rediscover such simple and clearly interpretable relationships as those above would be near impossible. This is a core difference between machine learning models and other scientific models, and one of the main reasons raised for why such ML models are less conducive to understanding.

Thus, from the inputs and outputs alone, it is not possible for the user to identify which features were significant for the given output, or to give any form of counterfactual explanation of the given output, contrary to what a more traditional model would have provided. Indeed, instead of providing the user with information of the output's dependency relations on the different input variables, all that can be inferred from a successful ML model is a certain amount of mutual information between the complete set of input features and the outputs. Note, however, that the absence of success does not imply the absence of mutual information, since there are numerous other reasons for why an ML model may fail in learning the desired mapping.

2.2. Explainable AI

2.2.1. Prediction without Explanation

The increasing use of machine learning models in a range of different fields, along with the heightened awareness regarding their opacity, has led to an increasing interest in establishing methods for explaining the predictions [3, 19]. Some have called for the use of more interpretable methods instead [17], but most agree that there is often a trade-off between interpretability and the accuracy and flexibility of Ml models, such that the more interpretable models tend to perform worse for complex problems, which is where we need them the most [7, 8]. Instead of using more interpretable models, then, the field of explainable AI aims to establish a set of additional methods for explaining the functioning of the opaque Ml models.

The range of different methods is vast, and there is some discussion regarding whether they even operate with the same notion of *explanation* [10]. We shall leave these concerns aside here, and limit the discussion and later investigations to a selected few of these methods.

2.2.2. Understanding and Explanation

There is a plethora of different accounts of what constitutes understanding and explanation out there, and no proper agreement. Luckily, the thesis of this project does not rest on any specific interpretation of these terms, so a somewhat vague definition will suffice.

Explanations are often taken to be counterfactual and causal. A counterfactual explanation tells you which features would have caused a different outcome, thereby highlighting the relevant features of the system. Such explanations are usually also causal, where it is the causal difference makers that are taken to be explanatory.

Importantly, however, is the difference between explanation and understanding. Explanations are usually taken to be a necessary, though not sufficient, condition for having understanding. Understanding also requires a more holistic grasping of the system as a whole, not just of individual outcomes [6]. For our purposes, it will suffice to include two requirements for understanding. Firstly, we adopt Knüsel and Baumberger's model-specific adaptation of de Regt and Dieks' [6] account of theoretical understanding, where they take the ability of the model user to qualitatively anticipate model outputs without performing calculations or running a simulation as a measure of understanding. Additionally, if the

user understands the model and/or the phenomenon modelled, he or she should be able to *explain* the model and/or specific outputs. Finally, understanding is not an all-or-nothing feature, but comes in degrees.

a. Understanding the model or understanding the phenomenon? Before elaborating further on the different methods, a quick clarificatory note on the object of explanation in these cases may be of use. The main goal of methods from XAI is to provide explanations of the ML model itself. However, this model is itself often used with the intention of understanding some target phenomenon from which the data was gathered. In such cases, the ML model is used as a mediator between the phenomenon and the explanation. Thus if the model has learned what we hope it has, it should somehow represent the phenomenon truthfully, such that the explanation provided by the XAI method does not only explain the model, but also the target phenomenon itself.

2.2.3. Methods and types of explanations

a. LIME This method was first introduced in [16], the acronym standing for *Local Interpretable Model-Independent Explanation*. This method provides explanations of the individual outputs in the shape of significance weightings for the different input features. Specifically, it works by taking a single prediction and its corresponding inputs. The algorithm then performs slight perturbations to the input variables to generate a second dataset. The perturbed inputs from this dataset are then passed through the original ML model, and the predictions recorded. The features are then ranked by how small perturbations in there values were sufficient in causing significant changes in the predictions, and those which caused the greatest change are considered the most important. This is similar to a counterfactual explanation, where the algorithm tells you which features would have to be changed in order to change the output. In the end, the model provides a significance score for each variable, which can be interpreted as a linear model of the ML model.

b. Shapley values Shapley values is similar to LIME in that it provides a significance ranking of the different variables, but it provides a more global explanation. It uses cooperative game theory (XXX how and why?). It tests different combinations of changes to the input variables, and calculates the average change in the predictions from changes in each of the input features, where simultaneous changes to multiple variables are also taken into account.

c. Symbolic Regression Symbolic regression is a method for identifying mathematical equations which fit a dataset. In this case, we use the package PySR (Symbolic Regression for Python) to approximate a mathematical expression for the ML model. This package uses a genetic algorithm to identify the most suitable equation, thereby providing a global explanation of the entire model.

2.2.4. Common Criticisms

There are three main criticisms commonly raised against such methods from explainable AI.

Firstly, they compute different functions the the original model, such that they are not truthful representations of the original model. In many cases the XAI algorithm provides one with a simplified function, for example linear, which is meant to represent the ML model. However, the original ML model will almost compute a non-linear function which is far more complex, and cannot be replicated using a linear function. Thus, the "explanation-function" is necessarily different to the original function, and will sometimes produce different outputs given the same inputs. For this reason, some argue, the model cannot be an explanation of something which it misrepresents and disagrees with. Babic *et al.* [1] go as far as calling the understanding provided by such methods "ersatz-understanding", claiming they merely provide us with an illusion of explanations and understanding.

Furthermore, there are cases where different XAI methods produce different and incompatible explanations of the same target model. This seemingly undermine their reliability, as we are left unable to decide which explanation to trust.

Secondly, some argue that the locality of the explanations provided by methods such as LIME and Shapley does not provide real understanding of the phenomenon, but only of specific instances of it [15]. Instead of providing us with an explanation of the global properties of the model, they only describe how the model works in a small region of the input-space. Underlying this critique is an assumption that there are some unifying features of the target phenomenon, such as a simple law of nature, which the model should have learned. Thus, they see it as a drawback that the XAI methods may produce very different explanations for datapoints which are close in feature-space, assuming that such discontinuities do not occur in nature.

Finally, in cases where the goal of the XAI method is not primarily to understand the ML model itself, but first and foremost the phenomenon modelled, it is essential that the ML model is at least reliable, and preferably also robust. We therefore need to be certain that we can trust the original model in order

to justify extrapolating the explanation provided beyond the ML model and onto the target phenomenon itself. This is not really an issue with XAI methods as such, but may or may not be an issue, depending on the final target of explanation.

a. XAI Models as Idealized models The latter two objections are the ones this project aims to solve. The first, however, I believe Fleisher [7] to have an excellent response to. He there makes a compelling argument for why the explanations provided by XAI are analogous to idealised scientific models, which there is a general agreement that that provides scientific understanding, regardless of their falsity. Fleisher points to three salient features shared by both idealised scientific models and XAI explanations to support his argument:

- Simplification: the model provides a simplified version of the target system, which is both interpretable and understandable to the user.
- Flagging: the model identifies and highlights features of the target system which are present, but negligible in the given context, e.g., by setting their value to zero.
- Focusing on specific causal patterns: the model suppresses some of the causal patterns which are present, so as to highlight certain patterns and their mechanisms in isolation.

Thus, if one believes that idealised scientific models, such as the harmonic oscillator or Snell's law, provides scientific understanding of their target phenomena, one should accept that methods of XAI provide true understanding of their target models.

2.3. Correlations and Causality

In very general terms, the aims of science are twofold. On the one hand, one aim to make accurate predictions, and on the other, to explain why these predictions are as they are. In the absence of any grand unifying theory, we tend to make do with specific theories for specified domains, and with models with even more restricted domains. As with Ml models, however, there is a goal of having models that are both reliable and robust, where, as before, reliability is the predictive power in the original domain, whereas robustness is the predictive power of the model in related domains. The question, then, is what features are needed of the model to ensure these properties, beyond predictive power on the original data used to construct it.

Assuming we have a dataset of observational data, as is commonly the case for machine learning research, we can imagine using a linear regression model to generate a reasonably simple and interpretable model. If the phenomenon from which the data was gathered is not too complex, there is a good chance that this provides us with a fairly good model. We are then justified in believing that the model will provide us with good predictions on new data points gathered from the same distribution, but may we also assume that the model will be predictively accurate for data gathered from different distributions? No. This is because such a model is purely correlation-based and does not assume any underlying causal structure of the phenomenon. It is what we call a *phenomenological* model. McMullin describes such model as follows:

> in general a phenomenological model appears to be an arbitrarily-chosen mathematically-expressed correlation of physical parameters from which the empirical laws of some domains can be derived. [...] From the purely logical point of view, there is no difference between a theory and a phenomenological model. Both can be axiomatised, from both; the desired empirical generalisations can be derived.
> But for the physicist there is a crucial difference between them. This difference can be put in one or other of two ways. The physical theory makes an assertion about a physical sub-structure which can account for the data; the phenomenological model makes no such assertion. Insofar as the latter goes beyond the descriptive level, it does so merely to obtain greater mathematical generality and not because there are physical reasons that suggest such a hypothetical extension to be appropriate. [11, p. 391]

Thus, if we want models which generalise beyond the specific dataset chosen, we need them to represent the underlying causal structure of the phenomenon [14] or the *causal capacities* of the different features of the system [5]. This will provide us with more stable and robust models which we may expect to apply more generally, at least in the absence of other causal difference-makers. Pearl, the founder of the field of causal inference, argues that we "expect such difference in stability because causal relationships are *ontological*, describing objective physical constrains in our world, whereas probabilistic relationships are *epistemic*, reflecting what we know or believe about the world. Therefore, causal relationships should remain unaltered as long as no

change has taken place in the environment, even when our knowledge about the environment undergoes changes." [14, p. 25]

Thus, if we want reliable, robust, and informative models of dynamical systems, models reflecting the underlying causal structure of the phenomena are preferable to those simply tracking the stable correlations between the system variables. This view has become increasingly widespread, and has lead to the establishment of an entire field of research, *causal inference*, dedicated to establishing the methods and empirical evidence needed to learn such causal relationships [4, 14, 18].

2.3.1. Structural Causal Models

The underlying causal structure of a system can be described in terms of a *structural causal model*, (SCM). The defining feature of such models is the *asymmetric* relations between the variables. Contrary to correlations, which are symmetric, causes and effects are not, as they come with a directedness. Such a model represents all the relevant variables in the system, along with a set of relations representing the direction of causal influence between the variables. Such a model can be expressed in either a set of equations, the *structural equations*, or in a causal diagram. An example of such a diagram is shown in 2, with its corresponding structural equations given in 1. The notation ":=" signifies the asymmetry of the relation, where the right-hand-side is the cause of the left-hand-side of the equation.

$$\begin{aligned} B &:= f_B(A, U_B) \\ C &:= f_C(A, B, U_C) \\ D &:= f_D(A, C, U_D) \end{aligned} \tag{1}$$

With adjacent nodes, A and B, when there is an arrow from A to B, A if referred to as the *parent* of B, and B as the *child* of A. If there is a further arrow that goes from B to a third variable C, then A is the *ancestor* of C, and C the *descendant* of A. Variables without any modelled parents, i.e., variables whose values are causally independent of the other variables, are called *exogenous*. Variables *with* parents are called *endogenous*. Finally, we always assume that there is some noise or background conditions which have not been accounted for in the model. These are grouped together in some random variables U, whose causal influence on the different variables are represented using dotted lines.

The notion of causality is closely linked to the notions of counterfactuals and interventions. This is because the causal structure is what determines the

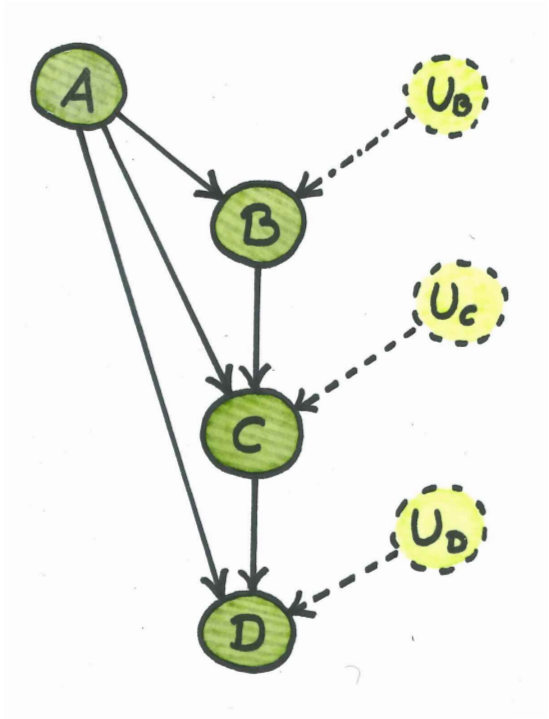

Figure 2: A diagram of a structural causal model. The variables *A, B, C* and *D* are the observed variables, whereas the variables in the dashed circles are their respective noise variables, which remain unobserved.

assymmetric dependencies between the variables, and these are most easily defined in terms of interventions and counterfactuals. Assuming there are no causal loops in the structure, a change in the value of a variable will lead to a corresponding change in the value of all its descendants, but not to any of its ancestors. In other words, given the model in 2, regardless of what actually happens, this model tells us that if we *had* intervened on variable B, this *would have* caused a change in the variables C and D, but not in A.

2.3.2. Learning Causal Models

The issue, then, is that we need some method for differentiating between correlational and causal relationships between measured variables. Given only observational data of the final state of variables, this is not possible. To illustrate this, three different causal diagrams are depicted in 3. Assuming only the variables X and Y are measured, using only the conditional probability distributions, there is no way

to identify the correct causal graph. This may not cause trouble as long as the variable Z is kept constant, but if any change occurs in Z this is likely to cause the model to fail. Z is here a *confounding variable*, which is an unobserved variable that causally influences the system, which is a common source of error in such modelling tasks. So how can we learn the right causal model?

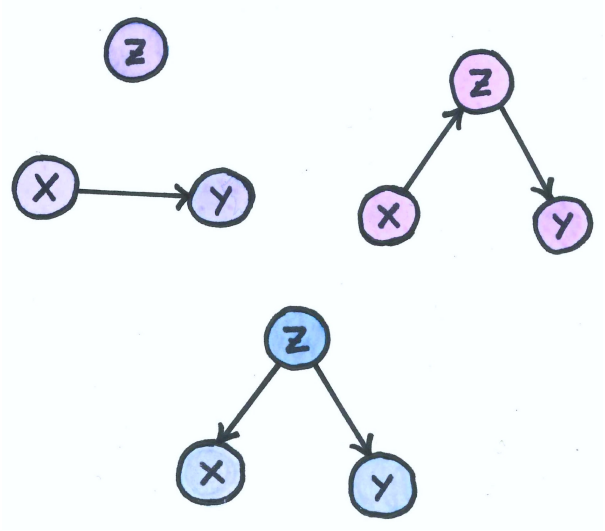

Figure 3: Three causal diagrams in the same Markov equivalence class. Given only observational data of the three variables, the three different causal structures may give rise to the same conditional probability distributions, so are indistinguishable.

The reliance on interventions in the definition of causal dependencies is suggestive of a method for identifying such relations. Pearl has introduced his "do-calculus" to formulate the conditions for causal inference in a more formal, mathematical language. Given a variable X, we denote the intervention where it is given the value x as $do(X{=}x)$. In general, it is assumed that all other variables are kept unperturbed, so that only one variable is intervened at at a time. By investigating the changes in the probability distributions of the other variables, it is then possible to identify the causal effect of X on the other variables. From this, Pearl defines the causal effect as follows:

Definition 2.1: Causal Effect

"Given two disjoint sets of variables, X and Y, the causal effect of X on Y, denoted either as $P(y|\hat{x})$ or as $P(y|do(x))$, is a function from X to the space of probability distributions on Y. For each realisation x of X, $P(y|\hat{x})$ gives the probability of $Y = y$ induced by deleting from the structural equations of X all equations corresponding to variables in X and substituting $X = x$ in the remaining equations." [14, p. 70]

Thus, given interventional data, one is more likely to be able to identify the underlying causal structure of the phenomenon, of course depending on the amount and type of data available. This can for example be done using measures such as the *average treatment effect*.

3. METHOD

3.1. Dataset

Each dataset was generated from a structural causal model. This defined the causal dependencies between all the variables in the dataset. Initially, no noise was included, such that the system was purely deterministic.

Three different datasets were generated based on the same underlying model, one observational and two interventional. In the observational dataset, all the exogenous variables were drawn randomly from uniform distributions, and the endogenous variables calculated based on this. In the interventional datasets, interventions were performed on one variable at a time. The values of all the ancestors of the variable intervened were then calculated as a function of the value set during the intervention, whereas the other variables were left unperturbed. Two different datasets were generated based on this: one were no information was included regarding the intervention, and another where the information about which variable was intervened on was included as a set of additional datapoints.

For now, only one model was used, as illustrated in 4.

A, B and E are exogenous variables, and are drawn randomly from a uniform distribution with given bounds. Initially, the system is deterministic, so no noise is included. The structural equations for the remaining variables are as follows

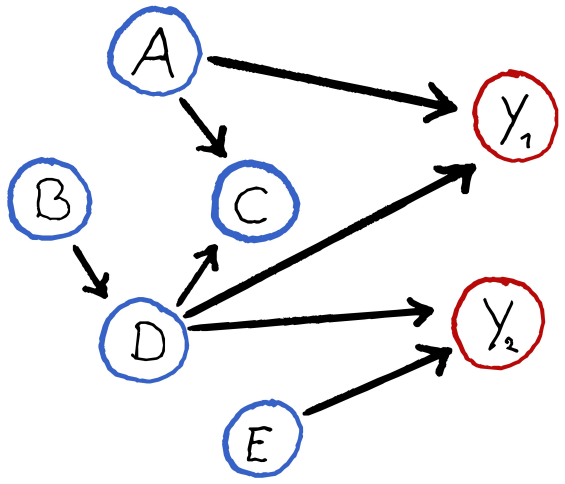

Figure 4: Model of the causal structure underlying the dataset used throughout. The input variables are illustrated with blue circles, the output variables with red circles. The arrows indicate the causal dependencies between the variables.

$$
\begin{aligned}
D &:= f_D(B) \\
C &:= f_C(A, D) \\
Y_1 &:= f_{Y_1}(A, D) \\
Y_2 &:= f_{Y_2}(D, E)
\end{aligned}
\tag{2}
$$

It might be useful to try out different functions with different degrees of complexity. Initially, the functions were set as follows

$$
\begin{aligned}
D &:= 2B \\
C &:= 5A + \frac{D}{A} \\
Y_1 &:= 3A^2 + D^3 + AD - D^2 \\
Y_2 &:= 4D - D^2 + \sqrt{E}
\end{aligned}
\tag{3}
$$

3.2. Network

A reasonably simple feed forward neural network was used. The only requirement was that the model performed well on the test-data from the same distribution, as well as on unseen data drawn from the same distribution but using a different seed. This was essential for the comparison with data from different distributions and domains to be reliable.

Of course, the state-of-the-art machine learning models for causal inference are far more complex than this. The reason for this choice of relatively simple network and datasets is to provide a proof-of-principle rather than to maximise the objective performance. By keeping it simple, the number of variables and potential sources of error are kept low, thereby making it easier to identify the mechanisms and important difference-makers in the datasets and networks.

3.2.1. XAI Methods

At least three different methods from XAI will be used. These include one local and two global methods, namely LIME, Shapley and PySR, which is a form of symbolic regression. See 2 2.2 2.2.3 for an outline of the different methods.

For the local method, LIME, the aim was to test whether the explanations were more stable for the models trained on interventional data. This was measured by comparing the variation in the explanations for different datapoints, e.g. using functional variance.

For the symbolic regression method, the question will be whether the explanations provided are closer to the true data generating functions when interventional data is used than when the model is only trained on observational data.

3.2.2. Observational Data

Depending on the results obtained for the interventional data, we might also experiment with some potential methods for extracting more information about the underlying structure just form observational data. In make cases, it might be either impossible to intervene on the system (for example for ethical reasons), or one might not have enough data to train the network. In fields where this is the case, such as social science, it is common to exclude potential causal relationships by conditioning on the different features. This sometimes makes it possible to differentiate between correlations and causation in the datasets. This is a process that involves certain considerations and background assumptions about the causal structure of the phenomenon.

As mentioned above, the performance of the model may be taken as a measure of the mutual information between the input features and the output features. Thus, by experimenting with including and excluding different combinations of input features for the same dataset and network, it should be possible to gain further information about the dependencies on the different input features by comparing the relative performance of these models.

3.3. Limitations

There are certain structures which the model will be unable to discover due to confounding variables.

---

[1] Boris Babic, Sara Gerke, Theodoros Evgeniou, and I. Glenn Cohen. Beware explanations from AI in health care. Science (New York, N.Y.), 373(6552):284–286, July 2021.

[2] Florian J. Boge. Two Dimensions of Opacity and the Deep Learning Predicament. Minds and Machines, 32(1):43–75, March 2022.

[3] Florian J. Boge and Michael Poznic. Machine Learning and the Future of Scientific Explanation. Journal for General Philosophy of Science, 52(1):171–176, March 2021.

[4] Stefan Buijsman. Causal scientific explanations from machine learning. Synthese (Dordrecht), 202(6):202–, 2023.

[5] Nancy Cartwright. The dappled world: a study of the boundaries of science. University Press, Cambridge, 1999.

[6] Henk W. De Regt and Dennis Dieks. A Contextual Approach to Scientific Understanding. Synthese (Dordrecht), 144(1):137–170, 2005.

[7] Will Fleisher. Understanding, Idealization, and Explainable AI. Episteme, 19(4):534–560, 2022.

[8] Julie Jebeile, Vincent Lam, and Tim Räz. Understanding climate change with statistical downscaling and machine learning. Synthese (Dordrecht), 199(1-2):1877–1897, 2021.

[9] Sanjeev Kulkarni and Gilbert Harman. An Elementary Introduction to Statistical Learning Theory. In An Elementary Introduction to Statistical Learning Theory, volume 853. John Wiley & Sons, Incorporated, United States, 2011.

[10] Zachary C. Lipton. The mythos of model interpretability. Communications of the ACM, 61(10):36–43, September 2018.

[11] E. Mcmullin. What do Physical Models Tell us? In B. Van Rootselaar and J. F. Staal, editors, Studies in Logic and the Foundations of Mathematics, volume 52 of Logic, Methodology and Philosophy of Science III, pages 385–396. Elsevier, January 1968.

[12] I. Neutelings. Neural networks. URL: `https://tikz.net/neural_networks/`.

[13] M. A. Nielsen. Neural networks and deep learning, 2018. URL: `http://neuralnetworksanddeeplearning.com/`.

[14] Judea Pearl. Causality: models, reasoning, and inference. University Press, Cambridge, 2nd edition. edition, 2013.

[15] Tim Räz and Claus Beisbart. The Importance of Understanding Deep Learning. Erkenntnis, 2022.

[16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. URL: `http://arxiv.org/abs/1602.04938`, `doi:10.48550/arXiv.1602.04938`.

[17] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature machine intelligence, 1(5):206–215, 2019.

[18] Bernhard Scholkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. Proceedings of the IEEE, 109(5):612–634, 2021.

[19] Sanja Srećković, Andrea Berber, and Nenad Filipović. The Automated Laplacean Demon: How ML Challenges Our Views on Prediction and Explanation. Minds and Machines, 32(1):159–183, March 2022.

[20] Michael Tamir and Elay Shech. Machine understanding and deep learning representation. Synthese, 201(2):51, 2023. URL: `https://doi.org/10.1007/s11229-022-03999-y`.